

5th
BIRTHDAY ISSUE

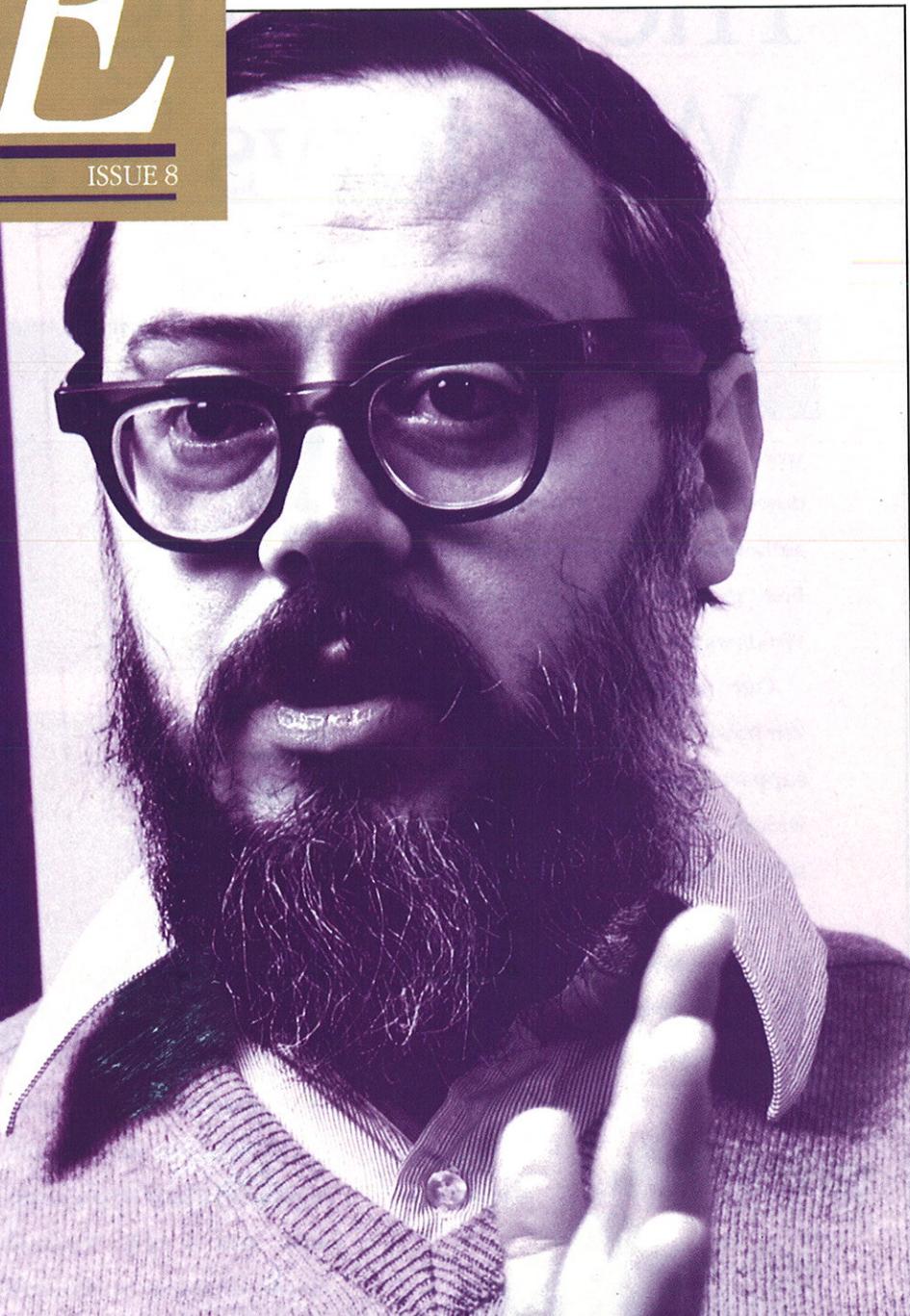
EXE

FEBRUARY 1991

VOL 5

ISSUE 8

The Software Developers' Magazine



*Who is this man? Here's a clue:
We focus on C and C++ this issue.*

*Yes, Dennis Ritchie, inventor of C,
tells Peter Collinson how it was.*

C++ and the post-ANSI C Standard.

*Eastern Europe Special #1:
The Romanians criticise TopSpeed C.*

*Eastern Europe Special #2:
A Pole on Polish computing.*

*Plus: we look at a new version of the
Smalltalk-80 development system.*

The Leading Light in Windows Training

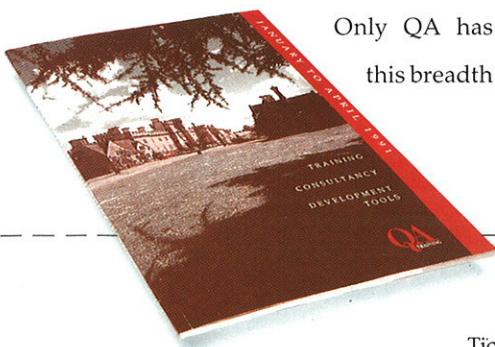


Training has long been recognised as the leader in Windows Training. The world's first company to offer Windows programming courses. The first authorised by Microsoft. And the very first to give technical training in Windows 3.

Our unique range of courses – Windows training for developers, support personnel and users – are widely acknowledged to be, quite simply, the best.

We also have worldwide involvement in consulting on many of the major Windows projects with corporate developers and software suppliers.

Only QA has this breadth



Introduction to Windows

Use Windows mouse and keyboard controls; Customise the Windows desktop; Work with Windows file and print utilities; Use Windows applications productively; Understand the theory of Windows. *2 Days.*

Supporting Windows

Install Windows and explore related issues; Configure Windows and applications; Use network, communications and print facilities; Understand Windows internals – memory, multi-tasking. *3 Days.*

Microsoft Windows Programming

Use the MS Windows SDK and tools; Write well structured compatible Windows applications; Understand the Windows API and Windows programming principles. *5 Days.*

Migrating Windows Applications to PM

Port Windows application to PM using the SMK; Understand structure and makeup of Windows and PM; Understand PM API including GPI; Understand object oriented structure for GUI code effectiveness; Aid migration with case oriented tools. *5 Days.*

of experience in every aspect of the Windows environment. Experience which is your assurance that a QA Windows course, is certain to maximise

your skills and effectiveness.

For more information, please complete and return the coupon today or call Samantha Trinder on 0285 655888.

Mail to QA Training Ltd, Cecily Hill Castle, Cirencester, Gloucestershire GL7 2EF, UK.

Tick here for full details:

<input type="checkbox"/> C	<input type="checkbox"/> OS/2	<input type="checkbox"/> Unix	<input type="checkbox"/> Windows
<input type="checkbox"/> Consultancy	<input type="checkbox"/> Object Oriented	<input type="checkbox"/> Networks	<input type="checkbox"/> Products

NAME _____

TITLE _____

COMPANY _____

ADDRESS _____

TEL. NO. _____



The Leading Light in
Windows Training

CIRCLE NO. 445

Editor:	Willie Watts
News Editor:	Dan O'Brien
Editorial Assistant:	Wendie Leek
Design & Layout:	Katerina Adams Mark English
Display Advertising Manager:	Sandra Inniss-Palmer
Senior Ad Executive:	Ed Butcher
Reader Services & Promotions:	Helena Adams
Publisher:	Jon Howell

General

.EXE Magazine is independent and not affiliated to any vendor of hardware, software or services. It is published by Process Communications Ltd, 10 Barley Mow Passage, Chiswick, London W4 4PH.
 Tel: (Advertising/Editorial/Production) 081 994 6477
 (Subscriptions) 044282 4501
 Facsimile: 081 994 1533 Telex: 8811418 SPACES G
 ISSN: 0268-6872

Subscriptions

.EXE Magazine is a monthly journal for software developers. It is available only by subscription, at a cost of £35 per annum (11 issues) in the UK (see insert card for academic and overseas rate). A subscription implies that this journal will be sent to the subscriber until one of the three expires - AG Macdonell. The magazine is published around the 26th of the month preceding the cover date. There is no January issue. To subscribe, please call 044282 4501, or write to The Subscriptions Manager, .EXE Magazine, 10 Barley Mow Passage, Chiswick, London W4 4PH. We can invoice your company or take ACCESS and VISA cards. In case of a query about your subscription, please call 044282 4501.

Back issues are available at a cost of £3.50 each - call our Chiswick office (081 994 6477) for a list of issues still in print. We also offer a range of magazine binders, disk holders etc - please call for details.

Editorial

Editorial enquiries should be addressed to The Editor, .EXE Magazine, 10 Barley Mow Passage, Chiswick, London W4 4PH. We welcome letters, opinions, suggestions and articles from our readers. If you are interested in contributing articles, please write to this office for a copy of our Contributors' Guide.

Information contained in .EXE is believed to be correct. If errors are found, we will endeavour to publish a clarification in the next available issue.

From time to time, we offer to copy the PC program code described in an article onto diskette for our readers. In this case, please send a blank, formatted disk with a self-addressed, prepaid mailer to the editorial address given above. We can copy both 5.25" and 3.5" disks.

The publishers can accept no liability for any consequences of using software distributed in this way.

Advertising

If you are interested in advertising in this magazine, please write to the Display Advertising Manager/Recruitment Sales Manager (as appropriate), .EXE Magazine, 10 Barley Mow Passage, Chiswick, London W4 4PH, or call 081 994 6477 for details of our advertising rates.

Pronunciation

The name of .EXE Magazine is pronounced to rhyme with 'not sexy magazine'.

Copyright

Material published in .EXE Magazine is copyright © Process Communications Ltd. Articles (or parts of articles) may not be copied, distributed or republished without written permission from the publishers.

Special Issue - C and C++**UNIX REGULAR**

UNIX irregular, actually. Peter Collinson is promoted to the front of the magazine, in honour of his interview with Dennis Ritchie.

14**C++ OR C9X?**

C++'s success has unexpected implications for C, says Derek Jones.

20**HERE ARE THE BUGS - TOPSPEED C**

Romanians Doru Turturea and Dan Somnea search for faults in JPI's C.

28**NAME MANGLING IN TURBO C++**

Borland's Alex Lane explains Turbo C++'s proprietary algorithm.

32**INSIDE THE INDEXES**

Ken Sawyer takes a look at the data files of Clipper, dBASE and FoxPro.

38**NOW WE ARE FIVE**

This month .EXE is five years old. We review past high and lowlights.

40**A NEW SMALLTALK**

Steve Cook has been testing Objectworks Release 4 from ParcPlace Systems.

43**NO EXCUSE FOR USER ABUSE**

What is an HCI consultant? John Harris explains why we need him.

51**MEETING THE DEADLINES**

Jim Cooling investigates the design of real time executives.

60**A VIDEO LABELLED 'X'**

The Editor switches into Nancy Banks-Smith emulation mode.

71**SOAPBOX**

Ray Jones lays the blame for the disappointing success rate of Structured Methods at the feet of the managers.

2**NEWS**

Zortech spreads across platforms like bindweed, the law descends upon software pirates and the Software Tools Show gets a new sponsor.

4**LETTERS**

We have two appeals: one from you and one from us.

10**THE THIRD SIDE**

George Ruscoe mourns the passing of the Mercury Autocode.

72**FROM OUR OWN CORRESPONDENT**

We begin a new regular series with Richard Sadowski on Poland.

76**CODE PAGE**

Conserve paper! Save trees! Dan O'Brien has an ecological program lister.

80**BOOKS**

Including virus wisdom from the man who foresaw the whole thing.

87**CROSSWORD**

Puzzle maestro Eric Deeson's latest cryptic teaser (4-4).

88**STOB**

Ham-fisted Verity battles with electricity.

96

Don't Shoot the Programmer

According to Ray Jones, it's the managers that need training in Structured Methods, not the programmers.

It's unfortunate but true that even after the great progress made in structured methods of analysis and design, and CASE tools, programmers still do badly. Software is frequently delivered late and below spec. In suit-speak, we have an 'ongoing software backlog'.

Yet there can be no doubt that the methods and tools developed over the last few years are effective. All the evidence points to this. The adoption of structured methods in software development improves quality, eases maintenance and shortens time scales. And the proper use of CASE tools to support the methods can only improve matters. So why don't more people use the methods? Or rather, why don't they use them properly?

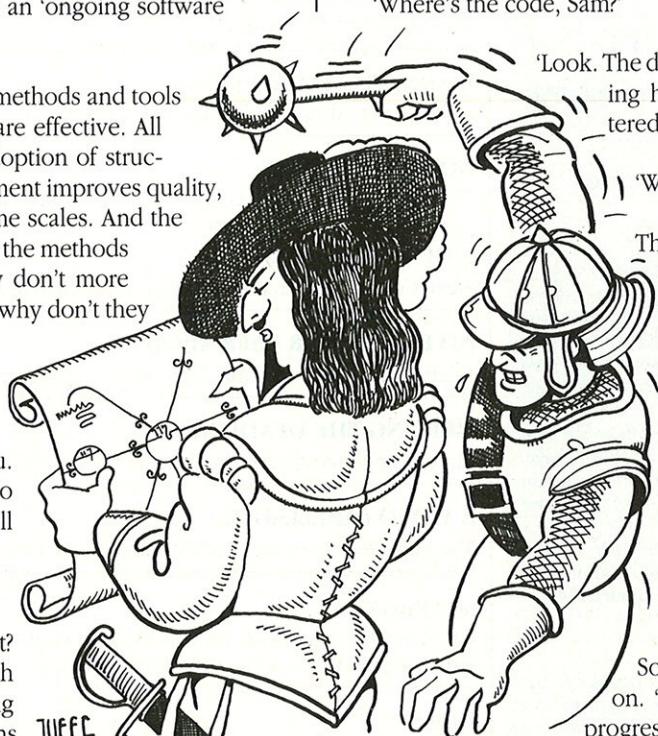
Hands up those of you who have been on structured methods courses - gosh! there are a lot of you. Now, hands up those of you who adopted those methods and are still using them six months after attending the course. Come on, be honest now. Aha! Rather fewer this time, I see. In fact, a lot fewer. Why is that? Why, when you were all fired up with enthusiasm at the end of the training course, did all your good intentions dribble away in the sand, like so many New Year's resolutions?

One of the reasons is that, when it comes to analysing the *real* requirements of *real* people and implementing *real* systems in the *real* world, it's not as easy as it seemed in the friendly and supportive atmosphere of the class-room, model solutions printed upside-down at the back. But then it never is. Nobody really expected it to be.

Another reason for the decline - and I believe it is the main one - is that when you get back to the office, the atmosphere is not friendly and supportive.

There you are, sketching elegant Data Flow Diagrams and exquisite Entity Relationship Diagrams, using your newly-learned skills to construct a Software Environment Model of the system that you will be implementing. But your project manager looks on, with the air of a Cromwellian watching them putting the undercoat on the ceiling of the Sistine Chapel.

'Where's the code, Sam?'



'Well, you see, with this diagram I will be able to plan the modules so that...'

'Where's the code, Sam?'

'Look. The data flows through the system starting here, where the invoices are entered by...'

'Where's the code, Sam?'

The project manager hasn't been on the Structured Methods course. He is too important to go on courses and anyway, he doesn't rate them after a disappointing punched card handling course he attended in 1975. He only understands one thing: that the result of your labour ought to be program code. He wants it now. Pretty diagrams might look impressive, but code is produced by sitting at a keyboard and typing.

So what do you do? The pressure is on. 'I need to be able to show some progress on this project at the next management meeting', says your manager. 'I need the first five modules coded by the end of the month'. Do you stick up for the methods of Gane and Sarson, Ward and Mellor or Michael Jackson? Do you quote the sayings of DeMarco or Boehme to support your case? Do you wave goodbye to the chance of an early bonus? Do you hecklers. You extract your digit, and code the first five modules by the end of the month.

It seems to me that this is a serious problem, and that it must be addressed. Training the analysts and programmers is fine - no, it is essential. But managers must also be trained in the modern methods of software production. They should know that the code is the last thing to be produced and they should be aware of the necessary stages in the development process that precede the coding. Only in this way can project managers successfully plan and monitor the progress of their projects. And only in this way, with proper management participation, will sensible and efficient development methods really begin to catch on.

EXE

Ray Jones is a Software Engineer who really does try to use proper design methods, honest.

THE C LANGUAGE

Microsoft C V6 is a complete rewrite with improved optimisation and a new Programmer's Workbench
 High C V1.6 has been considerably improved, with better Microsoft C compatibility,
 and new documentation

C COMPILERS

Avocet AVCOM 86C	PC-DOS	£210
Aztec C86 Professional	MS-DOS	£120
Aztec C86 Developer	MS-DOS	£175
Aztec C86 Commercial	MS-DOS	£265
CI C86 Plus v1.3	MS-DOS	£290
ECO-C88 V4.0	MS-DOS	£ 70
HIGH C (Metaware) v1.6	MS-DOS	£380
HIGH C 386 v2.3	PL386&MS-DOS	£640
Hi-Tech C	MS-DOS	£135
Lattice C v6	OS/2&MS-DOS	£175
Let's C v4	MS-DOS	£ 60
Microsoft C v6	OS/2&PC-DOS	£269
Microsoft QuickC v2.5	PC-DOS	£ 60
M'soft QuickC/QuickAsm	PC-DOS	£105
MIX C	MS-DOS	£ 20
MIX Power C & lib s'ce	PC-DOS	£ 29
Turbo C++	PC-DOS	£120
Topspeed C for OS/2	OS/2	£335
Topspeed C (std)	PC-DOS	£135
Topspeed C Extended	PC-DOS	£270
Watcom C v8 std	MS-DOS	£269
Watcom C v8 Prof	OS/2&DOS	£335
Watcom C/386 std	PL386&MS-DOS	£625
Watcom C/386 Prof	PL386&MS-DOS	£830
Zortech C++ v2.1	PC-DOS	£120
Aztec C86 Developer	CP/M-86	£245
Aztec C86 Personal	CP/M-86	£140
Hi-Tech C	CP/M-86	£150
Lattice C	CDOS	£380
Avocet C	CP/M-80	£210
Aztec C Personal 1.06D	CP/M-80	£120
Aztec Commercial 1.06D	CP/M-80	£200
Hi-Tech C	Z80 + CP/M-80	£ 99
Mix C	Z80 + CP/M-80	£ 35
Aztec C65 V1.05	Apple DOS	£200
Aztec C68K MPW C	MACINTOSH	£110
Aztec C68K C	MACINTOSH	£ 75
Aztec C68K/AM Developer	AMIGA	£175
Aztec C68K/AM Prof'snl	AMIGA	£120
Aztec C68K/ST Developer	ATARI	£175
Aztec C68K/ST Prof'snl	ATARI	£120
Hi-Tech C	ATARI	£ 99
Mark Williams C v2	ATARI	£110
Laser C	ATARI	£135
Prospero C	ATARI	£105

DISK COPYING SERVICE

We can copy files to and from 600 disk formats including CP/M, CP/M-86, MS-DOS, PC-DOS, ISIS, APPLE, SIRIUS, BBC, TORCH, APRICOT, HP-150, TRSDOS, DEC RT-11, IBM BEF, ATARI ST, AMSTRAD, MACINTOSH. Our charge is £10.00 + disk + VAT with discounts on small quantities and disks are normally despatched within 24hrs of receipt. For more information call us.

PRICES & DELIVERY

Prices do not include VAT or other local taxes but do include delivery in UK and Europe. Please check prices at time of order, ads are prepared some weeks before publication.

This page lists some of our products. Call us for a complete pricelist.

Order by phone with your credit card.

ASSEMBLERS

2500AD 80386 ASM v5	PC-DOS	£225
MS Macro-86 v5.1	OS/2&PC-DOS	£ 95
PharLap 386 DOS Ext SDK	MS-DOS	£370
PharLap 386 ASM/Linkloc	MS-DOS	£890
PharLap 386/VMM	MS-DOS	£240
Optasm	MS-DOS	£105
Turbo Debugger v2	PC-DOS	£105
2500AD Z80 ASM	CP/M-80	£ 70
SLR Z80ASM	CP/M-80	£ 40
SLR Z80ASM-PLUS	CP/M-80	£140
SLR MAC	CP/M-80	£ 40
SLR MAC-PLUS	CP/M-80	£140
SLR 180 (Hitachi)	CP/M-80	£ 40
SLR 180-PLUS (Hitachi)	CP/M-80	£140

Not all assemblers are supplied with a linker.
 Check before ordering.

THE C LANGUAGE

Microsoft C V6 is a complete rewrite with improved optimisation and a new Programmer's Workbench
 High C V1.6 has been considerably improved, with better Microsoft C compatibility,
 and new documentation

C CROSS COMPILERS

We supply 2500AD, Avocet, Aztec, Lattice, IAR and HI-Tech Cross Compilers hosted on MS-DOS and targeted on Z80, 6502, 6801, 68HC11, 6301, 6809, 7811, 8051, 8096, 68000 & 68020.
 Please call for information or advice.

C INTERPRETERS

Instant C/16M supports large programs on 286/386 machines.

C-terp V3.0X	PC-DOS	£165
Instant C/16M	PC-DOS	£700
Interactive C	PC-DOS	£195
Introducing C	PC-DOS	£ 85
Living C Plus	PC-DOS	£135
RUN/C Professional	MS-DOS	£110

FORTRAN COMPILERS

The latest Prospero Fortran now supports OS/2 & DOS.

A new version of Lahey's compiler with improved optimisation now here.

Lahey F77L v4.10	MS-DOS	£410
Lahey F77L-EM/32v3 ergo+PC-DOS	£665	
Lahey Personal Fort-77	PC-DOS	£ 79
FTN77/386 Fortran	PC-DOS	£765
RM/FORTRAN 77 v2.43	MS-DOS	£445
MS-FORTRAN 77 v5.0	OS/2&MS-DOS	£250
Prospero Fort 2.10	OS/2&MS-DOS	£240
Pro Fortran for GEM	MS-DOS	£ 80
Prospero PC Fortran	PC-DOS	£ 80
Utah Fortran	MS-DOS	£ 35
FS-Fort. (CGA & Herc)	MS-DOS	£ 34
Watfor 77	PC-DOS	£290

CLIBRARIES

DATABASE/FILEHANDLING

Btrieve for OS/2	OS/2	£315
Btrieve for Windows	MS-DOS	£315
Btrieve v5.1	MS-DOS	£315
XQL	OS/2, PC-DOS	£415
Xtrieve Plus	OS/2, PC-DOS	£315
C/Database T'lchest sce	PC-DOS	£ 39
CBTREE (source any C)	PC-DOS	£135
C-Index/Plus (source any C)	PC-DOS	£175
C-ISAM (L,MS)	MS-DOS	£215
C-tree (source any C)	PC-DOS	£210
C-tree Plus	PC-DOS	£350
Essential Btree (s'ce)	PC-DOS	£120
CQL s'ce	PC-DOS	£270
Faircom Toolbox Prof	PC-DOS	£610
Pro-C & Workbench v2	PC-DOS	£585
R-tree	MS-DOS	£160
db-VISTA DBMS Object	PC-DOS	£450
Lattice dBCIII (L,MS)	MS-DOS	£185
Lattice dBCIII+ OS/2&MS-DOS	£365	
SftFcus Btree&Isam (sce any C)	PC-DOS	£ 80

GRAPHICS

Enhanced Graphics Tlkt	PC-DOS	£170
Essential Graphics v3	PC-DOS	£215
GFX Graphics	PC-DOS	£ 80
Graphic v5 (CI,L,DS,MS)	PC-DOS	£290
graphics MENU/META v3	PC-DOS	£135
GSS Kernel system	OS/2, PC-DOS	£425
GSS Graph Dev T'kit OS2,PC-DOS	£425	
HALO v3 (MSC5,LAT,TC)	PC-DOS	£210
HALO Professional (C)	PC-DOS	£335
HGraph (MSC)	PC-DOS	£ 80
Ingraf (MSC)	PC-DOS	£175
MetaWINDOW v3.7b	PC-DOS	£150
MetaWINDOW/Plus	PC-DOS	£200
Professional Halo	PC-DOS	£335
TurboWINDOW/C (Turbo)	PC-DOS	£ 85
QuickWINDOW/C (QC)	PC-DOS	£ 85

SCREEN & WINDOWS

Vitamin C for OS/2	OS/2&DOS	£240
Vitamin C (source)	PC-DOS	£165
Power Screen (MS&TC)	PC-DOS	£105
Blaise View Mngr. (s'ce)	PC-DOS	£245
Lattice Curses	PC-DOS	£105
Entelekon Windows (s'ce)	PC-DOS	£105
Multi-windows (MS,L)	PC-DOS	£190
Panel Plus II (source)	PC-DOS	£270
Vermont Views (MSC,TC)	PC-DOS	£345
Greenleaf Datawindow	PC-DOS	£210
Greenleaf Makeform	PC-DOS	£ 70
C-Scape (+s'ce&Look&Feel)	PC-DOS	£345
Curses/PC w. s'ce	PC-DOS	£240

PROGRAMMING TOOLS

AdaCompilers	Algol Compilers
Assemblers & Libs	AWK
Basic Compilers	Basic Interpreters
Basic Utilities	Basic Libraries
BCPL Compilers	C Compilers
C Interpreters	C Libraries
C Utilities	Cobol Compilers
Comms.Libraries	Cross Assemblers
Database Libs.	Debuggers
Dis-assemblers	Editors
Engineers Libs.	Expert Systems
Forth	Fortran Compilers
Fortran Libraries	Graphics Libraries
Icon	Linkers/Locators
Lisp	Modula-2
Nial Interpreters	OPS 5
Pascal Compilers	Pascal Libraries
Prolog	Rexx
Risc	Screen Libraries
Smalltalk	Snobol

We stock many items for which there is no space in these advertisements.

C++ COMPILERS

Guidelines C++	PC-DOS	£370
Turbo C++	PC-DOS	£120
Turbo C++ Prof	PC-DOS	£200
Zortech C++	PC-DOS	£120
Zortech C++ Dev Edition	PC-DOS	£270
Zortech C++ OS/2 Dev Ed	OS/2	£360
Zortech C++ 386 Dev Ed	PC-DOS	£540

LIBRARIES

C++/Views (ZC+)	PC-DOS	£270
Greenleaf Comms (T&ZC+)	PC-DOS	£195
Object Vision	PC-DOS	£325
Zortech C++ Database	PC-DOS	£180
Zinc Interface Lib	PC-DOS	£150

CLIBRARIES

C0 Class Lib (MS,TC)	PC-DOS	£155
C0 Cl's Lb (MS)WINDOWS&PC-DOS	£155	
C/Utilities T'chest sce	PC-DOS	£ 39
C-Worthy v1.2 + forms	MS-DOS	£340
CodeRunner	PC-DOS	£105
PDKE1	PC-DOS	£ 75
Greenleaf Functions s'ce	PC-DOS	£125
G'leaf Super Functions	PC-DOS	£160
G'leaf Fin'cial Mathlib	PC-DOS	£210
Lattice C Comp Kits+ce	PC-DOS	£160
The Heap Expander (MS&TC)	PC-DOS	£ 65
C Tools Plus/6.0 (MS)	PC-DOS	£105
Power Search	PC-DOS	£105
Turbo C Tools v2	PC-DOS	£105
Essential C Utils s'ce	PC-DOS	£145
Entelekon Funct (source)	PC-DOS	£105
Resident-C (MS, L) (s'ce)	PC-DOS	£145
Silverware C EMM Lib	PC-DOS	£115
WKS Library	PC-DOS	£160

COMM'S LIBRARIES

C Async Manager V3s'ce	PC-DOS	£130
Essential Comms(L,MS,T)	PC-DOS	£175
Greenleaf Comms (source)	PC-DOS	£195
Multi Comms (L,MS)	PC-DOS	£190
Silvercomm C Asynch	PC-DOS	£145

SCIENTIFIC LIBRARIES

C Lang Sci Lib v2	ANY C	£160
Mathpak 87 (L,MS)	MS-DOS	£ 80
Science & Eng Tools	MS-DOS	£ 59

PROGRAMMERS UTILITIES

PC-Lint	OS/2,MS-DOS	£ 85
C-Documentor	PC-DOS	£195
C-Scan	PC-DOS	£195
Clear+ for C	PC-DOS	£140
Lattice Comp Companion	MS-DOS	£ 65



Debugger for NDP Compilers

Microway has launched a symbolic debugger for its NDP set of compilers for the 386/486. All the usual functions are included - source code display, single stepping, variable watching - together with support for 386 hardware debugging registers, and Intel and Weitek co-processors. The price is £250; Microway is on 081 541 5466.

BPS drops run-time licence

Bits Per Second Ltd has upgraded its Graphics Server product, mentioned here in November last year. V1.1 brings various technical improvements, such as the ability to open a Graphics Server window as a child window of an application, and a new pricing policy: BPS no longer charges for run-time copies of the product incorporated into applications. Graphics Server costs £275; the details are on 0273 727119.

Fox wins Ashton-Tate lawsuit

Ashton-Tate's copyright lawsuit against Fox Software, a manufacturer of a dBASE-compatible product, has been dismissed in a Californian court. The judge found that A-T had deliberately concealed the fact that dBASE was derived from a public domain program developed at the Jet Propulsion Laboratory. This deception was deemed to invalidate A-T's copyrights on dBASE. This result has interesting implications for the many other look-and-feel type lawsuits currently under way; in many cases the plaintiff is suing to protect his rights to software ideas which he himself had originally copied from a third party.

MS-DOS in ROM

Ever fancied your own personalised MS-DOS chip? Microsoft has released a developer's kit for the ROM version of MS-DOS, including a mini-licensing agreement to make up to five copies of the generated code. The pack consists of a kit for configuring and adapting the ROM image, reference manual, and documented IO.SYS source code. The system is aimed at small scale embedded-system OEMs, and can be ordered on 0734 391 123.

OS/2 Memory Utility

Masochists will love the new utility for PM from the OS/2 User Group. It displays in graphic form the amount of memory used by your OS/2 system. It's called Memmon (a pun on that other beast with an endless appetite, presumably) and costs £75 to members, £95 to others. Details available from Susie Hubber on 0285 655888.

Intel development kit puts Phar Lap's nose out of joint

Intel has taken the plunge into the software development market with a new 32-bit C compiler. The Intel 386/486 C Code Builder Kit contains a compiler, a source-level debugger, linker, libraries, utilities, MAKE and DOS Extender. The latter conforms to the fledgeling DOS Protected Mode Interface (DPMI) standard, and provides the user with full access to the full 4 GB address of the 386 processor. Significantly, there is no royalty for distribution on derived products. The package is not yet available in the UK, but retails in the US for \$695.

Phar Lap, the software house which made its name with DOS Extender technology, was less than chuffed by the announcement. Phar Lap chief Richard Smith told me: 'It seems strange that Intel, a chip company, is moving into the software market. One moment we're allies, all working together to produce the best possible tools to use the chips; the next thing we know, we are in competition with each other.' Regarding the DPMI standard, which is largely an Intel product, Smith noted that Intel had done a good job of bringing all the software developers together, but admitted that Intel's intentions in the software development market were not known at the time.

Commenting on a recent *Guardian* article which suggested that the introduction of DPMI servers would put DOS Extender producers out of business, Smith said: 'This just shows a misunderstanding of what we do. We are in the business of producing general development tools, rather than just DOS Extenders. The DPMI server represents just one quarter of what the developer uses; he'll still need our products to write his programs.'

You can get hold of Intel in the UK on 0793 696000. Contact Phar Lap in the US on 0101 617 661 1510.

Copyright Clamp-down

Several events in the last two months have signalled that companies throughout the industry are taking a tougher stance against piracy.

In December, the three largest UNIX vendors in the UK, Informix, SCO and Uniplex, combined forces with the pressure group FAST to combat what they describe as 'dangerous' levels of software theft amongst UNIX resellers.

Apparently, the bent dealers copy purchased software onto customers' systems, and then walked out with the same disks. FAST claims that up to 50% of current UNIX software is illegally installed in this way. The campaign to stamp it out will include a nationwide advertisement campaign aimed at resellers, and the publication of a safety check-list for buyers.

Meanwhile, in DOS-land, the Business Software Alliance, a group representing six of the largest PC software houses, has announced that four of their number are to file legal proceedings against Marconi Instruments Ltd for alleged software theft. The companies - Ashton-Tate, Lotus, Microsoft and WordPerfect - obtained court authorisation for an 'Anton Piller' order on the electronics firm in December. Under English law, this allows a solicitor to inspect without advance notice the premises of the company and obtain evidence which might otherwise be destroyed. Since the collection of evidence has always been the primary stumbling block of software theft prosecutions, this case, if successful, could well lead to further prosecutions.

At the same time as the Marconi prosecution was taking place, another two companies, Rhone-Poulenc Films and France Distribution Systems, were facing similar BSA-initiated litigation in France. The BSA is taking a particularly tough line in Europe, where software piracy is perceived to be more of a threat than in the States: a belief largely derived from national comparisons of the ratio of computer programs licensed to the number of machines sold. In America, the figure is 1.88 packages per machine - Europe averages around 0.6.

Dual monitor PS/2

Pity the poor PS/2 owner who wants to run CodeView in dual monitor mode. If he is determined to proceed, he must fork out for an IBM 8514/A video adapter (or perhaps an XGA would do the trick?) with hugely expensive high-definition monitor. Bournemouth-based Cebra Communications Ltd offers, for £495, a micro channel VGA board which cohabits with the one on the motherboard. As well as driver software to solve the CodeView problem, the company also bundles a Windows 3 driver. You can use this to perform that wonderful Mac trick of placing two monitors side-by-side and being able to drag a window off the edge of one, across the gap and onto the other.

Cebra has lots of other interesting video adapter cards for both traditional XT/AT bus machines and MCA - with enough of them you do things like hanging nine separate video displays off one PC. You can call the company on 0202 299048.

5 out of 5 hackers prefer other software protection methods to Hardlock E-Y-E®



What hackers dislike...

Hardlock E-Y-E was designed using cryptographic principles. It took the experience and know-how of Germany's No. 1 in software protection and the leading edge technology of a US semiconductor company to create the ultimate software protection tool. Hardlock E-Y-E is based on a custom chip featuring secure algorithmic response rather than simple bit swapping or counting schemes.

What software developers like...

Hardlock E-Y-E combines all the features software developers require in a single product: algorithmic response to provide security and an optional non-volatile memory to allow custom configurations. FAST Electronic has made implementation of Hardlock E-Y-E in your software easy. Use HL-Crypt to protect .EXE or .COM files, or incorporate high level language interface routines in your software. The algorithm parameters and the contents of the memory can be programmed in seconds using our Crypto-Programmer card. This unique card guarantees that no one else can burn your original codes. Simply plug the card into any PC slot and start up your own Hardlock E-Y-E workshop.

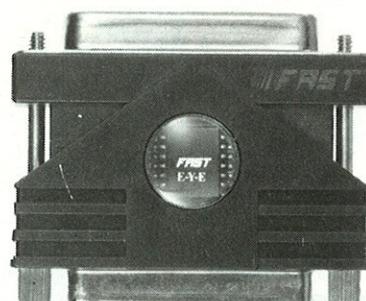
What your customers will like...

Hardlock E-Y-E allows unlimited backup copies. The device is shipped with the software for the user simply to plug into the parallel interface and forget.

Daisy chainability, outstanding reliability (no battery is needed), and the most compact High-Tech design ensure that your customer will accept Hardlock E-Y-E.

What your accountant will like...

Hardlock E-Y-E needs no factory coding. This ensures optimum delivery schedules and stock flexibility. Revenues will go up as software piracy and multiple usage are prevented. Despite its wealth of features, Hardlock E-Y-E's prices remain competitive.



Hardlock E-Y-E
programmable, algorithmic response
and memory option – all in one.

...As more and more software developers, customers and accountants appreciate the Hardlock E-Y-E device, hackers like it less and less.



Order your demo unit today. Contact Magnifeye,

235-239 Walmer Road, Walmer Studio #6, W11 4EY, Telephone 071 221 8024, Fax 071 792 3449.

CIRCLE NO. 448

Stats News

A new product from those Top Maths Boffins at NAG, who have released a graphics package and prototype for their industry standard maths libraries. NAG is a non-profit academic organisation with a reputation for high quality mathematical software. They've left graphics alone until now, however. The new package is called PV-WAVE:NAG, and is a collaboration with the American firm Precision Visuals. It's only available on the Sun-3 and 4 workstations and VMS (why do we keep mentioning VMS this month?) at the moment, although a PC version is being mooted. More details from NAG on 0865 511245.

Zortech broadens its range

Zortech Ltd has introduced a range of products based on its new 32-bit C++ compiler engine. The DOS 386 developer's edition contains a full 32-bit compiler, together with specially adapted 386 versions of the Zortech debugger, workbench (including an editor with a multi-gigabyte capacity), a standard library with source, a set of tools and the Flash Graphics library. The package is based on Phar Lap's 386/DOS extender; the SDK for this needs to be purchased separately. The DOS 386 version joins the existing 'C++ for MS-DOS and Windows' and 'OS/2 C++ Developer's' products, making Zortech's offerings for the single-user PC pretty comprehensive.

Also due for release by the time you read this is C++ for SCO UNIX. As well as the compiler itself, Zortech has ported its workbench and Flash Graphics so that they can be used on the main console of SCO UNIX machines, and it is now working on getting the debugger to work in the same way. Other flavours of UNIX are also planned for the near future.

As if this wasn't enough, Zortech now has its eye on the Macintosh. C++ for the Mac is currently in Beta-test, with a planned release date in late March. This product will integrate with the Macintosh Programmers Workbench and the MPW debugger, so can be used as a direct replacement for MPW C and C++.

CASE:W for Glockenspiel CommonView

A new version of the popular application generator tool for Windows, CASE:W, has been announced, this time for users of Glockenspiel C++ and the CommonView class library. Programmers can now produce code in the familiar Notifier/Control/View class style. CASE:W for CommonView shares the same interface and features of the rest of the CaseWorks series, including the useful regeneration feature, which allows CASE:W source output to be hand tweaked, and then re-incorporated into the system.

CASE:W itself can run on a 'minimal Windows system', but it will, of course, need the Glock kit and the Windows SDK to compile. It costs £795, and is available in the UK from Systems Resources, Tel: 0203 220246.

Prices: the DOS 386 compiler costs £599.95. To this you must add the Phar Lap kit, which Zortech sells at £299.95. The SCO UNIX package is also priced at a shilling under £300. Contact Zortech on 081 316 7777.

Objects for Windows

Microsoft has released a preliminary specification and code of a new set of DLLs designed to support object-based communication between Window applications. The approach taken by the new system will be an extension of the current DDE technique of allowing many applications to control different elements within a single document. The Object Linking and Embedding specification (Microsoft wants it pronounced 'Olé!') will be included in standard Windows in the next release and will also be implemented on the Macintosh System 7.0 when it arrives. It has the support of Aldus and Lotus among others. The latest versions of Microsoft Excel and Lotus Notes are the first two applications to use the new system. Copies of the specification are available from Microsoft on 0734 391123.

Object Management Group Publishes Guide

The Object Management Group, the OOP Standards group, whose patrons include AT&T, Apple, Borland, Whitewater and HP, has released its Object Management Architecture Guide, a proposed standard for Object Oriented environments. The guide steers clear of specifics, but does present important descriptions of the OMG's proposed interfaces between differing systems, and a description of the OMG's evaluation procedures. OMG isn't a standard making body yet, and the omission of IBM and Microsoft from its ranks implies that it might never be, but the publishing of these guidelines has partly shaken off OMG's talking-shop image. Both Apple and Dataflex have joined the group following the guide's appearance. Copies of the £30 document can be obtained from OMG on 0332 513855.

IBM takes Smalltalk/V

IBM has licensed Digitalk's Smalltalk/V for OS/2. Digitalk will be working with IBM on Smalltalk/V enhancements, including a set of visual programming tools, increased AD/Cycle integration and SAA compliance. A Windows 3 version of Smalltalk/V should be available now. Digitalk's products are distributed in the UK by Cocking and Drury (071 436 9481).

Assembly Offer

The Software Construction Company is running a promotion this month (February). Anyone purchasing the popular, but pricey £255 Spontaneous Assembly code library from TSCC will also receive a free copy of MASM V5.1 (usually retails for £115) and a copy of Using Assembly Language from Que (£25). TSCC is on 0763 244114.

Coherent UNIX Update

Coherent V3.1 is the newest release of Mark Williams' cheap UNIX-compatible operating system. New features include COM3 and COM4 support, a RAM disk and a vi/ex clone called Elvis. Not in there yet is a C compiler that produces anything larger than small model programs, but they're working on it. Coherent V3.1 is still \$99.95. Mark Williams can be contacted in the US on 0101 708 291 6700.

APL for Windows 3

*STSC has a new version of its APL*PLUS package for the PC. Features in version 10 include multiple instances in Windows, new mouse and PostScript printer support, and, for those APPLers who would stoop so low, cross-linking with Microsoft C 5.1, Turbo C V2.0 and Microsoft FORTRAN V5.0. APL*PLUS PC is distributed in the UK from Cocking and Drury, 071 436 9481. STSC itself is on 0753 831541.*

Quintec Prolog

A good month for the dropping of runtime licences, this. Prolog-maker Quintec Systems (0865 791565) has announced that it will no longer charge a licence fee on applications created with Quintec Prolog. The package, currently at V2.0, is available for UNIX workstations and PC running Windows 3.0.

Inmos and Ready Systems

Ready Systems Corporation, producer of the VRTX real time operating system, has entered into partnership with transputer maker Inmos. The first fruit of the new relationship will be a port of VRTX32 onto current and next generation (H1) transputer families.

ideaList

ideaList is a free-format database for the PC from Blackwell Scientific Publications Ltd. According to its promotional literature, it can find all references to any string in the database within milliseconds - the technology is similar to that used to drive mainframe information systems and CD-ROM databases. The package costs £225, ring 0865 240401 for the details.

Ada Still Free

Thornbrook, whose offer of a free Sun or VMS Ada Compiler suite to deserving academic institutions was mentioned in December's issue, has been so impressed with the interest generated that it is extending its offer to the 31st of March. To recap, the offer includes a full compiler, run-time system, debugger and documentation, and would usually sell for \$15,000 to \$70,000. John MacGregor awaits your calls on 0993 831333.

C into AutoCAD

Autodesk Ltd has shipped Release 11 of the well-known AutoCAD drafting program. The new feature of particular interest to developers is the AutoCAD Development System, which provides an API through which third parties can hook in their own program modules, which must be written in Metaware High C 386. ADS supplements the earlier AutoLISP system. AutoCAD costs £2500; the Advanced Modelling Extension is £400 extra. Autodesk may be contacted on 0483 303322.

Documentation Standard

The BSI has published a draft British Standard to cover the area of software documentation. Document number 90/66493 DC, 'Recommendations for the design of user documentation for software products for text and office systems', is intended to guide technical writers as to how to present information in a most effective way. The draft standard costs £6.25 to BSI subscribing members and £12.50 to others. The BSI is on 0908 220022.

Born out of need?

Milspec Systems Ltd, the Ada compiler company, has produced a RAM manipulating program. PLUS96K works by stealing 96 KB of graphics RAM from a PC's EGA/VGA board, increasing available 'conventional memory' from 640 KB to 730 KB. Naturally it doesn't work with graphics programs; but at £29.99 including VAT and postage it seems quite reasonably priced. Milspec's number is 0203 670770.

A new look for Software Tools

Software Tools '91, the trade show which caters for the software developer, is to be held at the Wembley Exhibition and Conference Centre from 11th-13th of June. This year, for the first time, it is to be sponsored by .EXE Magazine.

It has already been noted elsewhere that computer trade shows seem to be growing more and more dreary as time passes. Where once the aisles thronged with excited chatter and the crowds queued up to see the latest product unveiling; now a desultory trickle of bored, grey men wander around the lonely spaces, sometimes pausing at the odd booth to make a half-hearted attempt to strike a deal on the purchase of 30/off 386s.

We have determined that Software Tools '91 will buck this trend. There will be 'Spot the Bug' prize competitions, where punters riding a line of PCs will compete to get a bit of code back on its feet, and Software Marathons, where teams from the likes of Microsoft and Borland will race to complete set projects. There will be technical support booths, where you will be able to confront in person the bloke who is always so evasive on the phone. There will be lectures and presentations, covering all the latest GUIs and 32-bits and OOPs and OSs. There will be Really Important Gurus, who, their beards glistening in the conference hall lights, will clear up the deepest computing mysteries with a few well-chosen words. There will be a shoal of little stalls, where the software dealers will display their wares like fruit at Sainsbury's, and you can buy anything from a real time operating system or a sizzling C++ compiler to that interesting little utility that you've always thought would come in handy, but couldn't be bothered to send away for. The big software companies, the 'major players', will be there as well; parading their latest stuff and giving us glimpses of what is in the pipeline. And there will be lots of people: chatting with the vendors, trying out the software, discussing a few beers over pints of foaming algorithms.

At least, we hope that all these things will happen. We already have Borland, JPI, Microsoft and the Software Construction Company with us, which is a dandy start, and we have a few projects under way, but we need your help and support. If you would like to find out about booking space at Software Tools, you should contact Julia Dunford at Blenheim Online Ltd (081 868 4466). If you have a cracking idea for something we should do at the Show, please contact us at the .EXE editorial office (but be quick - time is short). And, of course, we hope to see you on the day.

**Protection for MegaBasic**

MegaBasic is a PC-based version of that hallowed language. Developed in the US, it started life as North Star BASIC (remember the North Star Horizon?) and these days, according to Murdoch Mactaggart of UK distributors The Software Centre, finds many of its fans among ex-mainframe and mini programmers, typically working in engineering, process control and the like. The particular feature which attracts, according to Mactaggart, is the ability to get at low-level features; for example, it is possible to manipulate CPU registers directly.

A new version of the interpreter, called Extended MegaBasic (rather than Mega²Basic, as you might have guessed), runs AT class machines or better in protected mode - thus gaining access to the 16 MB address space. The 80x86 segmented architecture restricts string variables to 64 KB; but string arrays can now be dimensioned up to 1 MB in length and numeric arrays to 500 KB. The language is upwards-compatible with non-extended MegaBasic, and the interpreter contains its own DOS

extender - no further purchases are necessary. License-free run-time modules enable you to sell on your applications at will.

The Extended MegaBasic development system retails at £375. If you want to compile your programs, you will need a copy of the standard MegaBasic compiler (£275) which, in a way that I don't fully understand, is able to produce both normal and protected-mode programs depending on which library you use. Phone The Software Centre on 0935 862 609 for the gen.

dBASE IV for VMS

Ashton-Tate is now shipping the VAX version of dBASE IV V1.1. To launch the new product at Comdex, Ashton-Tate bravely offered to port any existing DOS application to VMS, UNIX or Macintosh, apparently successfully. As well as running as a DOS-code interpreter, it will also support VAX Rdb/VMS databases using NAS SQL. UK prices start at £969.00+VAT for a single user package and £4090.00+VAT for a four-user licence, and the distributor is P & P Micros and Technology Plc (0925 830404).

db_VISTA III™

Database Management System

FEATURES	YES	NO
Single and multi-user available	✓	
Relational B-tree indexing	✓	
Network database model	✓	
Multiple database access	✓	
Referential integrity	✓	
Transaction processing	✓	
Automatic recovery	✓	
Record and file locking	✓	
RAM resident		✓
Relational Query and report writer	✓	
Total database redesign/restructuring	✓	
C compilers: most supported	✓	
C ++ compatible	✓	
Operating Systems: VMS, Ultrix,	✓	
UNIX, BSD, SunOs, XENIX, QNX,	✓	
MSDOS, MS Windows and Macintosh	✓	
OS/2 compatible	✓	
LANs: Netbios and Appleshare	✓	
Read and write WKS, WK1 and DBF files*	✓	
Source Code available	✓	
Training courses available	✓	
Run-time Royalties (Absolutely NOT)	✓	

*using WKS Library

The db_VISTA III™ database development system is intended for use by the professional C applications developer. db_VISTA III is written in C and provides a complete set of sophisticated development tools that feature:

- High speed access to large or complex data
- Mainframe functionality
- Portability to any C environment
- Royalty-free run-time distribution
- Source code available in C.

db_VISTA III provides both Relational and Network model technology for programming flexibility. Retrieve a record using the relational keyed access method and all related records can be immediately available using the network database model.

db_QUERY provides SQL-like access to db_VISTA III, callable from within an application, db_REVISE allows restructuring of live databases.

db_VISTA III is fast. Access times for retrieving data are largely independent of the size of the database.

db_VISTA III is portable. Develop an application in any environment and your database access code can be ported without change to any other supported environment. Currently, db_VISTA III supports MSDOS, MS Windows, OS/2, Macintosh, QNX, XENIX, UNIX System V and Berkeley, Ultrix and VMS.

db_VISTA III is a product of Raima Corporation and has been sold to thousands of C programmers in over 50 countries. db_VISTA III is fully supported in the UK and Eire by Systemstar Ltd. Comprehensive training is available together with bespoke system development. At Systemstar we offer C-scape 3™ screen interface library with Look and Feel™ from Oakland Group Inc to provide db_VISTA III programmers with a complete application development environment.

For more information about db_VISTA III, db_QUERY, db_REVISE and C-scape 3 call Systemstar in Hertford on (0992) 500919.



SYSTEMSTAR
L I M I T E D

1-3 PARLIAMENT SQUARE HERTFORD SG14 1EX
TELEPHONE: (0992) 500919 FACSIMILE: (0992) 554261

CIRCLE NO. 449

Letters

We welcome short letters on any subject that is relevant to software development. Please write to The Editor, .EXE Magazine, 10 Barley Mow Passage, Chiswick, London W4 4PH. Unless your letter is marked 'Not for Publication', it will be considered for inclusion on this page.

PS/2 troubles

Sir,

I would welcome any solutions or suggestions from your readers on a problem I have encountered with the 'protected mode' on 80286 and 80386 machines. I use the BIOS interrupt 15h function 89h to switch to protected mode, whereupon I search extended memory for data previously put there by calls to Microsoft's XMS driver, HIMEM.SYS. This driver allows MS-DOS programs access to extended memory. On a standard AT, this seems to work perfectly. However, on a PS/2 Model 50 (80286 machine, MCA), the same program hangs after a few calls to the protect-mode search routine. This happens even though I disable all interrupts while the program is in protected mode. Maybe someone out there knows something about either the MCA or the new BIOS that I don't?

Joe Tierney
London NW1

Problem solver

Sir,

Disappointing isn't it? After half a century of technical progress, the IT industry is still discussing the relative merits of various programming languages.

If object-oriented programming (OOP) had been a panacea, SIMULA would have swept the board in 1967. Similarly, despite the greatest hype yet perpetrated, Fourth Generation Languages (4GLs) still fail to fulfil the promises made on their behalf. Worse, the computer education of our children concerns itself with teaching specific coding languages. Concentrating our attention on such issues avoids the point and amounts to sublimation activity.

Surely it is past time to recognise that our industry exists to solve problems. Everything is secondary to that aim.

It seems reasonable to me that, in order to provide a solution, I must first describe the problem I wish to solve. From the problem description I should then be in a position to derive a solution. Indeed, I wish to be able to derive a number of solutions, one for each potential implementation choice. Further, if the requirements of the problem change (as they will), I would expect to change the problem description and to maintain the solution by regenerating it.

What is needed then is a specification language in which I can write problem descriptions which themselves can be transformed into an implemented solution. Preferably, the transformation should be automatic. The coding language of the implementation is incidental and can be based entirely on convenience. A minimalist might say that I only need four operators, ASSIGN, ADD, IF, and GOTO. However, productivity will no doubt be improved by an ability to build a library of the descriptions of the small building blocks common to many problems. I should not expect such building blocks to approach the complexity of the

whole problem, as the re-useability of a description is in inverse proportion to its complexity.

Where should I look for such a specification language? I do not expect to find a universal cure but I do expect the various system development methods to share common cause with me. In practice, I use the Jackson System Development (JSD) method with whatever system software and coding language proves convenient (most recently, Pascal, COBOL and ADA).

PG Rule
Senior Software Engineering Consultant
LBMS PLC

Hybrid Greek

Sir,

'Sieve of Erastophenes?' (.EXE Nov '90, Page 8)
Eratosthenes was the maths.
Aristophanes was the plays.

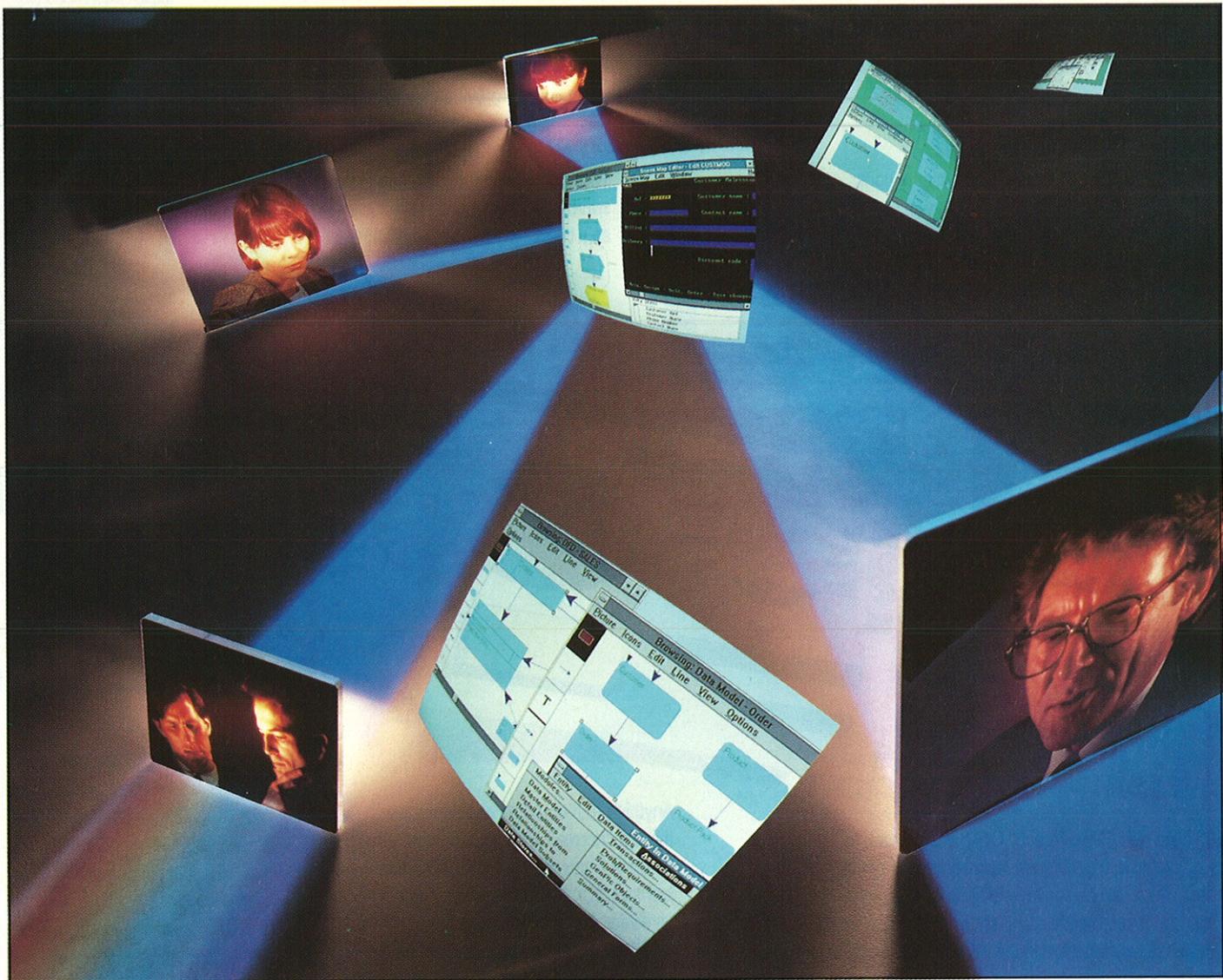
Mike Kinghan
Principal Analyst Programmer
AT&T ISTEI
Oxford

Is anybody out there?

Have .EXE readers lost the art of letter-writing? The flow of letters into our Editorial office has been most disappointing over the last few months, and we have now reached the point where Something Must Be Done.

Starting next month, and at enormous expense, the writer of the best (as judged by the Editor) letter that we print on these pages will be rewarded with an elegant, stylish .EXE T-shirt. Readers who belong to the Campaign Against T-shirts may nominate some other .EXE trinket of approximately the same value - an elegant, stylish magazine binder, an elegant, stylish disk holder or whatever; see your elegant, stylish .EXE Directory for details.

So if you have got an interesting coding problem, or you have discovered a plague of bugs in an expensive compiler, or you tried typing in one of our programs and it formatted your hard disk (only joking), or you think that one of our writers is an unrequited prat who should not be let within 200 yards of a computer, let alone allowed to try to *program* one, then reach down the Basildon Bond and drop us a line. We look forward to hearing from you.



LBMS Systems Engineer.

Multi-user CASE to boost your team's productivity.

It's here! Practical CASE automation that takes team productivity to a new high . . .

LBMS Systems Engineer. Built on the LBMS pedigree, this multi-user windows-based tool offers leading-edge technology to support the rapid application techniques you want to use.

So why multi-user? Systems Engineer networks your team to provide concurrent and immediate access to shared information. The resulting accuracy and consistency are the keys to rapid progress in application development.

How does Windows help? Windows™ 3 is a breakthrough in performance and usability. Systems Engineer fully exploits this natural

desktop approach. As a result, multiple techniques can be used side-by-side to achieve the ideal decision-making context for quality designs. Systems Engineer transforms the PC into a powerful developer's workstation by integrating CASE with your choice of Windows™ tools (word processing, e-mail, project management etc.). For added flexibility, an OS/2™ version will be ready when you are.

As an IBM AD/Cycle™ vendor and partner to other leading industry suppliers, we offer an integrated set of CASE and methods solutions that can be matched perfectly to your development environment.

So that you gain the very best from our CASE tools and methods, we back you with the level of training and support that's appropriate to you. Our aim is to transfer the skills and experience of the best people in the business to your organisation.

Get LBMS integrated CASE solutions working with your team. Phone Elizabeth Baxter now on 071 636 4213 for a demonstration, more information or details of our CASE technology seminars.

Evelyn House 62 Oxford Street London WIN 9LF
Tel: 071-636 4213 Fax: 071-636 2708

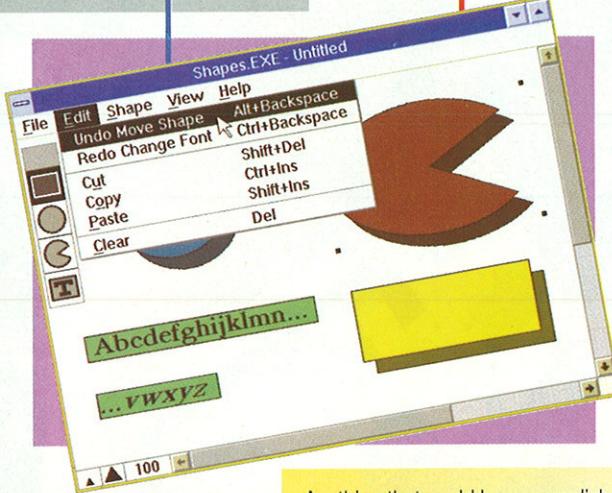
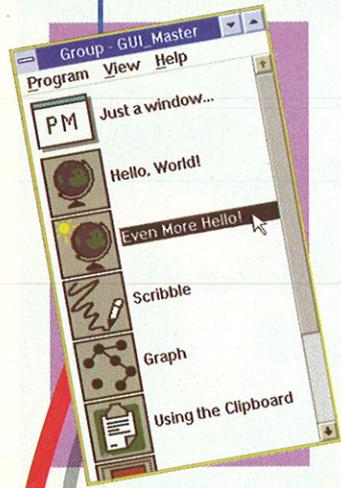
LBMS

Trademark ownership. OS/2™, AD/Cycle: International Business Machines Corporation. Windows™ : Microsoft Corporation.

CIRCLE NO. 450

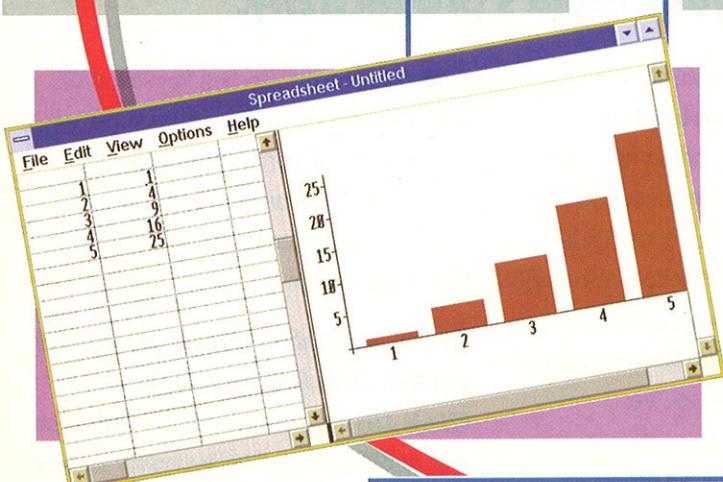
In the example programs you can see how the *GUI_MASTER* takes care of all the standard CUA functions like

- Scrolling
- Runtime error handling
- Printing
- Undoing/Redoing the most recent user actions
- Text editing
- Standard mouse interactions such as dragging and sketching
- Archiving objects
- Standard dialogs such as open, save as, fonts, color picker
- Using the clipboard



The *GUI_MASTER* (Class Tree for C++) enables you to specify different worlds, different representations of the same data which can be shown in different parts of the same window. If you modify some of the data in one world this is automatically reflected in the other world.

If you use *GUI_MASTER* your application is, as a matter of course, structured around the Data/World concept. An excellent basis for GUI applications.

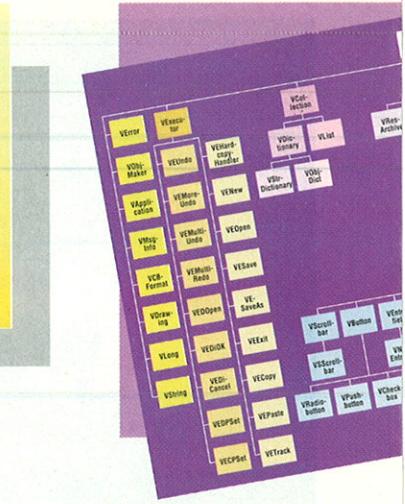
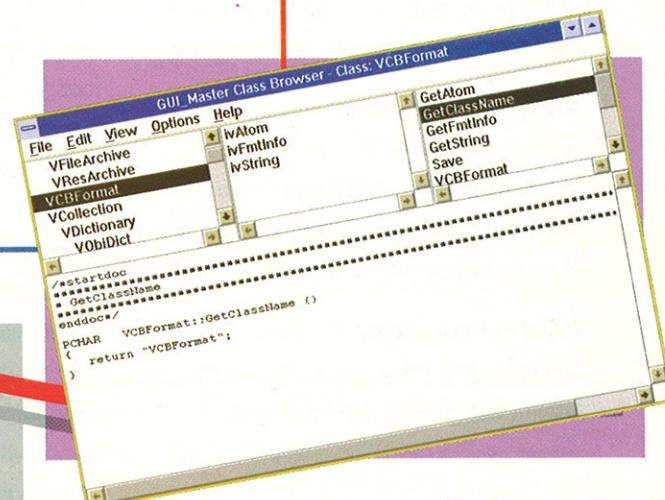


The class browser is of crucial importance to enable you to view programs as collections of classes and not as a set of source files. It enables you to examine code made by others, and to understand the class structure of that code, thereby promoting the concept of 'the programmer as reader' and stimulating the reuse of software.

With our browser you can

- browse through the source code of any C++ program
- launch the editor of your choice to modify your C++ programs
- quickly locate any method in the Class Tree

The 7 example programs range from simple to fairly complex. Together with the cookbook we provide they help you gain insight in all the possibilities of the *GUI_MASTER* (Class Tree for C++). They come complete with fully annotated source code, so you can even use them as a basis for your own applications.



Effectivity MS Windows Presentation

If you are building sophisticated C++ applications why not benefit from our *GUI_MASTER* (Class Tree for C++) to give you

- a head start in building GUI applications
- a GUI toolkit consisting of 85 reusable classes, together forming a well-tested and fully functional 'standard application'
- platform independence between MS/Windows and OS/2 Presentation Manager

The product offers

- 7 example programs
- a Class Tree
- a C++ Source Browser

in developing or OS/2 Manager programs

- extensive documentation
- an Interface Builder

To use *GUI_MASTER* you need

- a C++ compiler (we suggest using the Zortech C++ 2.1)
- OS/2 Toolkit or MS/Windows Software Development Kit

The *GUI_MASTER* (*Class Tree for C++*) enables you to focus on the real application parts without having to invest your valuable time in building the GUI parts.

\$ 495.= OS/2 Presentation
Manager version
available now
\$ 545.= MS/Windows
3.0 version
(May '91)

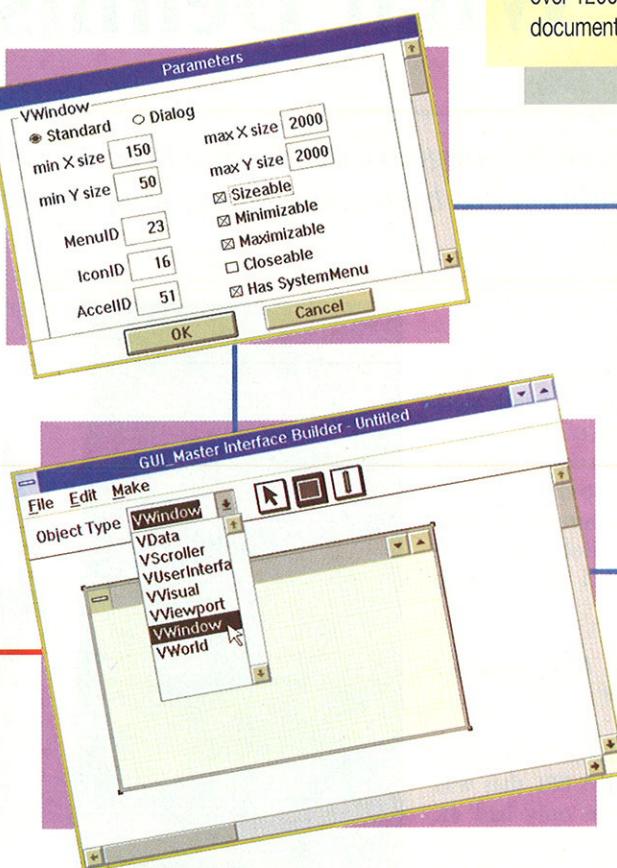
Vleermuis Software Research is an independent research organization with nearly 100 researchers covering all aspects of 'application enabling'. In the past three years VSR spent over 40 person-years on OO development on a broad spectrum of commercial workstations. VSR publishes the Journal of Software Research 4 times a year. Evaluation copies can be obtained via the address on the order form.

You probably don't want to invest your valuable time in reinventing the GUI wheel. We have therefore included 85 classes in our *GUI_MASTER* (*Class Tree for C++*), containing a wide range of GUI building blocks.

You may want detailed reference documentation.

The *GUI_MASTER* (*Class Tree for C++*) comes with

- a Cookbook, containing detailed how-to information
- over 1200 pages of class reference documentation, fully indexed of course



The Interface Builder allows you to specify all the properties of every visual object. It then generates for you

- the necessary source code
- the resource specifications
- and even the make file

The Interface Builder helps you create your own classes. It treats your classes in the same way as the other classes in the Class Tree.

With the Interface Builder you create the necessary windows and dialog boxes for your application simply by painting them on the screen. The objects created in this way already have a default behavior, which can easily be changed. By painting your objects on the screen you implicitly create a fully functioning program.

CIRCLE NO. 451



To order the *GUI_MASTER* (*Class Tree for C++*) mail to

Please print or type all information.

Vleermuis Software Research bv

P.O. Box 2584
3500 GN Utrecht
The Netherlands

Fax: Int'l. +31 30 31 04 26

Name: _____

OS/2 Presentation
Manager version

of copies Total amount

MS/Windows 3.0 version

@ \$ 495.= _____

@ \$ 545.= _____

Airmail costs included (delivery within appr. 3 weeks)

Delivery within a week by
DHL Worldwide Express

@ \$ 50.= Yes/No

Total

Payable by certified cheque or by MasterCard



CreditCard number: _____

Card Exp. Date: _____

Signature: _____

What Dennis says

*If any programming language can be said to be the general-purpose standard, it is C.
Peter Collinson has been talking to the man who invented it, Dennis Ritchie.*

How did C start?

On the PDP-7 UNIX system everything was written in assembly language. Doug McIlroy did an implementation of TMG, which is a compiler writing system done originally by Bob McClure. It's a top-down parser, although much more programmed than that. Doug McIlroy and Bob Morris did the original Multics PL/1 compiler in TMG.

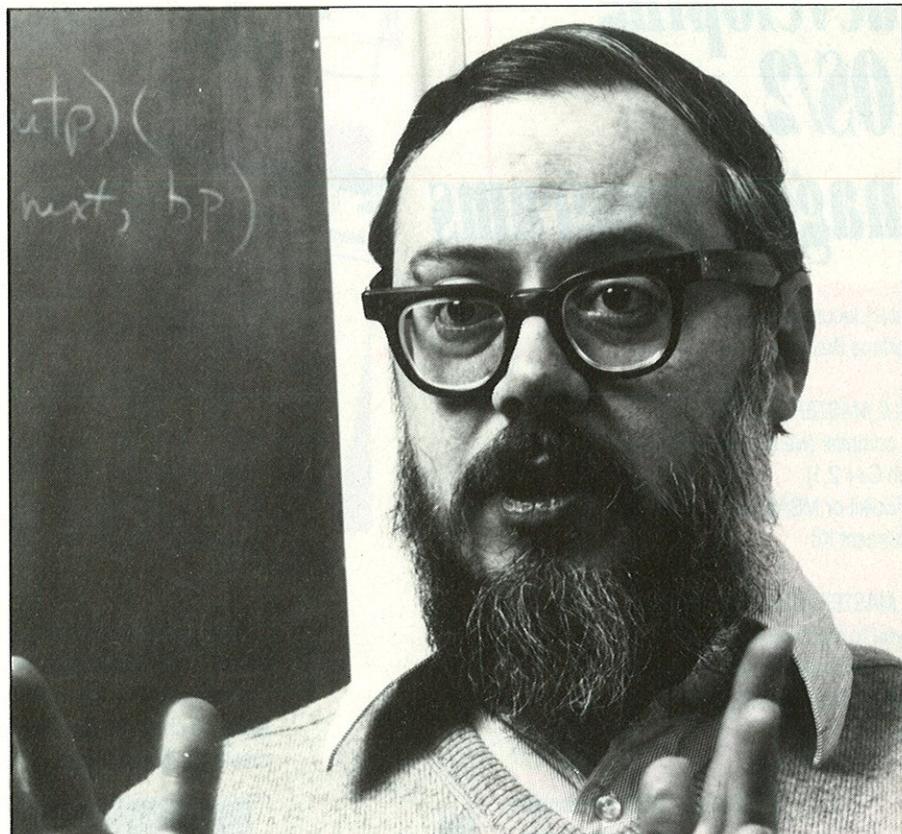
What was the main language of Multics, was that PL/1?

It was PL/1. When we were on it, it was an early dialect of PL/1 called EPL that was essentially PL/1. Since we only had 4K words of memory for user programs on the PDP-7, PL/1 was clearly out of reach. Doug did TMG for the PDP-7. That was the year that Ken Thompson decided that we couldn't have a serious system without a FORTRAN compiler, so he sat down to write a FORTRAN compiler using TMG. This lasted about a day. Instead what happened was B. B was essentially a cut-down and modified version of BCPL.

Had you been exposed to BCPL?

At the time I came into the Labs, almost the very first thing that I did was to get hooked up with Rudd Canaday who was the person who had ported the Martin Richards' BCPL compiler from CTSS to the GECOS machine, the Honeywell GE-635. At the time he was moving the compiler to Multics. I got involved in Multics here by working on the BCPL compiler. This was an early dialect of BCPL; Martin Richards himself continued to change the language after we grabbed it.

BCPL had a strange way of doing external linkage, called the global vector. It didn't have some of the control statements. This is one of the reasons why switch statements in B and C differ from the BCPL switchon statement. I don't believe he



If you don't know what C is, then you have accidentally picked the wrong magazine out of the pile, and should put .EXE back at once. The C language was originally developed, together with the fledgeling UNIX operating system, at AT&T Bell Labs in the early 1970s. Dennis Ritchie is C's designer, and is also co-author (with Brian Kernighan) of 'The White Book' - The C Programming Language - which remains the definitive C reference. Peter Collinson interviewed him at Bell Labs, where he still works.

had things like endcase at that time, so the languages differ in the way they come out of case blocks. I added continue later in B. Besides some obvious syntactic differences, some of the control structures were different just because it was an early version of BCPL.

BCPL was around and we had written some noticeable programs in it. On the first UNIX system, B was a new simpler language based on BCPL. It was an interpreter and didn't produce machine code; it produced an intermediate code.

How did you bootstrap it, was it in assembler?

It was first written in TMG, and later it was bootstrapped. In fact, the development of the language was amusing. The compiler was always pushing against the size limit in the machine. Ken would add something to the compiler and there would be a painful period when it was very hard to add something. But then he would include the new construct in the compiler - and that would make it smaller again allowing more new things to be added.

B was really the first high-level language that was used on UNIX. A few things were written in it. There were two implementations: one was this ordinary interpreter generating an intermediate stack based language. There was also a thing called *vb*, a Virtual B, that was a software paged version of the something allowing 4K word segments to be paged in and out by the interpreter. Whenever we had a program that got too big to fit into memory, *vb* was used.

By the time we got the PDP-11, only a few things had been rewritten in B. One of the first programs was the PDP-11 assembler. There was also a version of *dc*, the desk calculator which was the first program that ran on the PDP-11. It ran stand-alone before there was an operating system.

Did you put in a standard DEC operating system on the PDP-11?

We got the PDP-11 very early. It came during the summer of 1970. Only the processor and memory arrived, there was no disk. It was all paper tape software, you loaded programs with paper tape, there was no operating system as such. The first UNIX for it was written cross-assembling from the PDP-7 using the PDP-11 assembler that was written in B.

UNIX came up in two stages. Ken got it going before there was a disk. He divided the memory up into two chunks and got the operating system going in one piece and used the other piece for a sort of RAM disk. To try it out, you would load this paper tape that first initialised the disk and then load the operating system. There was a *cp*, a *cat*, and an *ls* actually running before there was a disk.

Once the system was really there, there was a regression. The B version of the assembler was pretty slow and that was rewritten in assembly language. I guess there were a few new things that were written in B. One of the earlier ones was the thing that did the expansion of stars and whatnot in filenames: the *glob* command. This stood for 'global' for reasons that escape me; it's not very sensible. The command did shell expansions. When the shell recognised it had a magic character in an argument, it would call this command to do the work. The code wasn't built into the shell.

First came B...

What was B like?

Apart from tiny syntactic differences, B was exactly like C, except there was only one type that was *machine-word*. In that sense

the semantics were the same as BCPL. There were no structures. The control structures were there. B did have some development of its own. I believe that the *for* statement got added somewhere in the middle of B development.

He decided we couldn't have a serious system without FORTRAN, so he sat down to write a compiler

C was developed by a process of evolution from B. The most important reason for change was that because of the typelessness of B, there was no really convenient way of directly dealing with bytes or even machine words. There really were two different kinds of things on the PDP-11, characters and integers. Even though the machine didn't have hardware for it yet, it was clear that there was about to be floating point and that would add at least two new different types. There would be one and two byte integers with four and eight byte floating point numbers.

The way that you deal with different types in BCPL is to have different operators. We had a version that did floating point and this was expressed by having different operators. Similarly there was a special function that was used for characters. This whole approach seemed to go OK on the word oriented machines, but it really broke down on the byte oriented machines.

The other thing was that it was fairly clear that once you began to compile code and produce decent machine language, the word orientation of the language was going to cause inefficiencies. The rule of the BCPL language is that you add one to a pointer to get to the next object, which is a word. On a byte oriented machine you are more or less forced to represent the pointers as counters of words, and to use them you have to shift them to produce the actual machine address. This seemed artificial. It was OK in the interpreter, it was glossed over as an interpretive overhead. But it was at least artificial and clearly would be inefficient.

The first thing for C was to introduce types to cope with these problems. This would allow efficient access to the hardware of the machine. Also, with the introduction of typed pointers, pointers contained addresses of an object of a type. Arithmetic on the pointers could be scaled appropriately.

Once the type structure was in C, I wanted to make a compiler so it would be fast enough to be able to be used for real things. At the time we were very much afraid of the compiler technology for the PDP-11, which was a two address machine that none of us had seen before. A single instruction, like a *MOV* instruction, could have two general operands. We were all used to single address machines, and for a while everybody was nervous about whether it would be easy to compile code for the machine.

I then heard about the PhD thesis of a guy who was working at Bell Labs at Indian Hill. The crucial idea of that was to have templates that matched bits of the parse tree. The templates have associated code fragments that would be inserted in the code stream. The code generators were written using this idea.

The two basic notions were: first, whether something is addressable. That is: whether or not some fragment of tree can be turned into an address mode instruction. Second, it was not addressable, whether or not it matches another table entry. If you have an assignment with operands that are both addressable, then you can produce a single *MOV* instruction.

There were four sub-tables: one for ordinary expressions that produce a value in a register; one for pushing a value into the stack; one for doing an expression only for its side effects, where you didn't care about the value; and finally, one for doing an expression in a conditional context where you generated a branch.

Anyway, that sort of worked.

Structures

Where did the idea of structures come from?

I think that I added structures to versions of BCPL on Multics. I think that the Multics version had structures in the PL/1 style, namely with level numbers. That was fairly clumsy. The nested structures probably came from Pascal.

I wanted to add them to C. The other thing that went on at that time was a real resolution of the issue of arrays and pointers. To

go back a little: a vector or an array in BCPL was precisely a single cell containing a pointer to some space. If you said

```
let avec = vec 10;
```

`avec` became a pointer to the base of these 10 cells which were otherwise anonymous. In other words, the notion of the interchangeability of arrays and pointers that exists in C was even more explicit in BCPL. There really were no arrays, only pointers. The pointer was really in existence, and had a cell associated with it. You can add one to `avec` and it becomes a pointer to the next cell.

This view of things persisted into B, and was exactly the same in the early stages of C. The declaration of an array of 10 things produced a cell that had type 'pointer' internally and pointed to anonymous space. At this point, I began to think about how to put structures into the language. I realised that this notion didn't work with structures.

In BCPL there was an opportunity to create the pointer to the vector for both ordinary external static stuff and automatic variables. In the case of the structure, there was no point at which you would think that this pointer would get initialised properly. If you allocate a new structure containing an array, then exactly when does the array pointer get set up properly?

Similarly, structures were intended to be mapped onto memory. It seemed certain that if you were trying to describe some memory layout that included an array, then you would need some place to put the pointer.

It was at this point the crucial notion of C appeared that makes it different from almost any other language. Instead of having the array pointer to be explicit and live in storage associated with the array, the pointer should be created when you use it. In other words, there should be a conversion between the name representing an array and the pointer to the base of the array.

It turned out that with this idea, everything continued to work. All the programs had the same meaning as they had before, except that an explicitly defined cell became a pointer to the base of the array. It made things more efficient too because whereas before you were forced to pick up a pointer to do address arithmetic, now you could do things more directly. Also, it solved the structure problem. Suddenly, there was no instantiated pointer that we needed to fit into a structure or worry about how to initialise.

In some ways, it was the structures that made C such a good assembler, if you see what I mean...

Yes, that was our experience too, quite explicitly. In 1972, Ken Thompson sat down to write the operating system in C. At that point it was still in PDP-11 assembler. C did exist and did have a compiler by this time. He started, but he finally gave up. The first reason that he failed was that we was not able to work out the fundamental coroutine stuff, how the processes hand off to each other. That was just a conceptual problem, trying to work out the famous 'you are not expected to understand this' part of the code. That was just the first solution to that problem - but he hadn't even gotten that far at that stage.

When we first got the PDP-11, you loaded all your programs with paper tape

Perhaps equally important, there were no structures in the language. Therefore, representation of all the various tables - things like the file table, the inode table and the process table - were all quite artificial. He had to do all these using a BCPL technique: having a pointer and a bunch of constants, and subscripting using the constants which represented the fields in the table.

The trouble with that is that you cannot differentiate the types of the words. You can have a structure which has a character pointer, but all type information is lost when you use the base pointer plus subscript. So you cannot do anything sensible with the contents of words, and all the work on setting up types is wasted. Maybe you need to have parallel tables, one with all the characters and one with all the integers etc. All of this was a mess and just didn't work. By 1973, structures had been added and that was the spur that we needed. When we started again to write UNIX in C, the structures turned out to be very important.

In some ways, it has taken quite a long time for structures to become full citizens.

That is partly because the White Book definitely said that structure assignment is not

in the language described by the book. On the other hand, it stated explicitly that structure assignment might be permitted in the future. Quite soon after the book appeared, the PDP-11 compiler had structure assignment and pcc had it. It became available and, although it was documented, it was never formalised in anything like a language description. It didn't always get into other people's implementations or there were slight differences in the implementation. For example, whether or not a function can return a structure that you can use as a member of, like this:

```
val = (structfn()).member;
```

where `structfn()` returns a structure with member `member`.

If there had been a second edition of the White Book in 1980 or so, at least some of the confusion that lead to the ANSI committee wouldn't have occurred. Things like the question of unsigned characters, structure assignment, the unique structure names - member names within structures do not have to be unique across all structures, they only have to be unique within their own structure.

Array problems

*I think that I have always found that somehow the history of how you invented arrays and structures has meant that people become confused. I mean the way that the address of a vector base is `vector` and also `&vector[0]`; and also that `vector[0]` and `*vector` return the same result.*

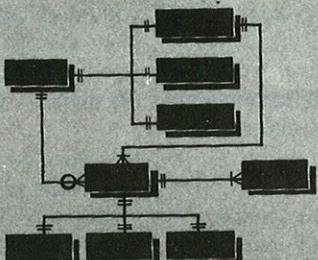
It does seem to be confusing. Perhaps that's because it was not explained correctly from the start. Although Brian Kernighan and I have had a chance to try to explain it, we may not have succeeded in making it obvious. Also, we haven't been helped by compilers, which have often been quite loose in what they allow. They often don't do anything like as strict enough type checking as they should, often not really checking the existing type structure of the language. So people get confused because they can try something which apparently works, and they are never told they have really broken the rules - that they are only being allowed to do this as an indulgence.

For example, this is how I feel about the business when pcc allowed ampersand to be placed in front of array names. It complained, but still allowed it. If the rules had been strictly insisted on by the compiler, so when people tried something they had immediate feedback saying 'this is wrong'; then they would have learnt the correct language a lot more easily.

In fact, for analysis and design, it's the best at any price. And here's why.

Recently a "Big Eight" accounting firm surveyed over 500 major MIS sites using CASE tools and concluded that the cost of CASE is the primary barrier to its initial use. And at a time when many companies are cutting to the bone to remain competitive. Even more disturbing is this same firm's conclusion that most of the available CASE products are too complicated to use. CASE products? Too complicated?

So, when *PC Magazine* recently judged



including: Chen and Information Engineering data

Functionality? The Workbench is loaded with it. Enough to support the needs of an entire LAN-based project team as well as a single analyst or designer. And because the Workbench is a full-function analysis and design tool, it helps increase productivity, improve system quality and reduce costs in both of these crucial development phases.

The Workbench supports well known and thoroughly endorsed development methodologies in-

items from diagrams, so that they can be defined in detail — even to the pseudo code level. The repository can be accessed from diagrams with a click of the mouse.

We offer a powerful prototyping tool, which, when added to the Workbench, enables you to "preview" applications before writing a single line of code.

The Workbench outputs are truly presentation quality. Standard support is provided for IBM PC graphics and HP laser printers, as well as additional support for PostScript, Encapsulated PostScript and GEM metafiles.

"Low Cost?" Absolutely. Workbench module prices start at £ 695.

Affordability, after all, is a virtue in any product, and especially so in one that keeps its promises.

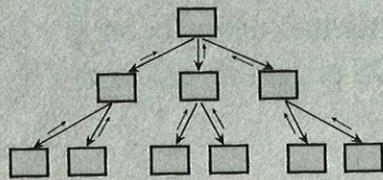
"Visible Analyst Workbench is the best of the lower-priced CASE packages."



January 30, 1990
Visible Analyst Workbench

Visible
Analyst
Workbench

to be "the best value available today" for its unmatched price/performance characteristics and awarded our product a coveted EDITOR'S CHOICE, we had good reason to be proud, but not complacent. Because while thousands of people are benefitting from the Workbench, many more thousands who could and should be aren't, simply because they mistakenly equate "low cost" with "lack of functionality."



UK Distributor:

THE SOFTWARE CONSTRUCTION CO. LTD.

Telephone (0763) 244114 Facsimile (0763) 244025

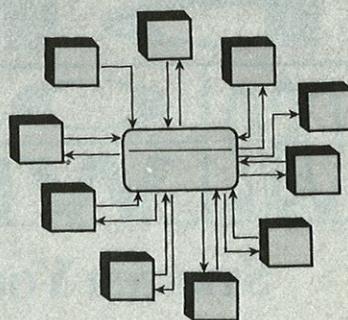
©1990 Visible Systems Corporation. Visible, Visible Analyst Workbench, Visible Solution, The Visible Analyst and Visible Systems Corporation are registered trademarks of Visible Systems Corporation.

*The Visible Analyst Workbench.
The best at any price.*

modeling (entity relationship models), Yourdon/DeMarco and Gane & Sarson process modeling, and Yourdon/Constantine design. Your development work will benefit from engineering discipline, structure and standards!

"Industrial strength" features? The Workbench has plenty. For starters, it features the most flexible graphical "front end" available in any CASE product today, which includes support for custom symbols and diagrams, reusable graphics, automatic "boilerplates," and the like. The Workbench also includes an extensive set of methodology rules, which check and validate projects for level balancing, completeness and logic flow, and provide usage warnings, statistics, complexity ratings, and more.

The Workbench contains a comprehensive on-line data repository, which automatically captures all methodology-related



V4.0 Pricing:

Visible Analyst Single User	£ 695
Visible Analyst Workbench Single User	£ 2460
Visible Analyst Workbench Single User with Prototyper	£ 3090
Visible Analyst Workbench LAN version (3-Node System)	£ 6995
Each additional node of concurrent access	£ 1295



Visible Systems
CORPORATION

CIRCLE NO. 452

Another problem is the business of rewriting the types of formal parameters to routines from 'array' to 'pointer' because it was just confusing. The specific reason that the rewriting is allowed had to do with the difference in notation at this moment when structures and the modern notion of arrays were both created. The original notation for a pointer declaration did not use stars, it used empty brackets, so instead of

*PP

you said:

PP []

and that was indirection through a pointer. In the original language, an array was this cell containing a pointer to anonymous space. The notation for just the pointer was consistent with that. It was the pointer with no storage shown by the empty brackets. The survival of the bracketed parameters was a fossil of this notation.

But I agree that people do seem to have difficulty understanding all these things, and partly it may be inherent because it's an unfamiliar way of doing things, it may be an unnatural way of doing things although it is consistent. Partly it's that the language and its compilers allow a certain amount of slop that encourages this confusion.

Do you feel ANSI got rid of any of this?

Well, the trend in the last 10 years has been towards stricter checking of the rules that already exist. In structure references in the original language, you were allowed to use any pointer to the left of an arrow. The compiler simply said, 'OK, I don't care what type that pointer is, I'll believe what you have told me'. That had been tightened up by the time of the White Book, although not all the compilers enforced it.

ANSI has not actually made that much of a difference in terms of tightening up things, but they have certainly made the rules more explicit. But I cannot think of any aspect where they have made them stricter. In some sense they have made them looser because they put in the explicit void *, the universal pointer that you can coerce to and from without having to give an explicit cast. It's much better to do it explicitly, than happen all the time 'for free', as it did in the original. The original language used char * rather than unsigned and all sorts of nonsense like that.

There has definitely been a process of strengthening the type-checking, which is not only a matter of diagnosing things but also of having a clearer understanding of the

semantics of what is actually happening. As a result, there are some quite explicit holes left in the type structure and the possible checking that you can do. The language is actually much stricter and more easily checked than most people suspect. In general, the compilers have been lax about doing things like subscript range checking and a variety of other things. Of course, there do remain the explicit escapes like casts of pointers.

It is possible to write interpreters that do enforce the rules of the language successfully. It turns out that if you represent pointers internally as a pointer to the data plus a lower and upper bound, then the rules are sufficient to propagate the upper and lower limits so you can do subscript checking correctly. We have an interpreter that does that, and a number of other people do too.

To be continued.

EXE

Many thanks to Dennis Ritchie for generously taking the time to give this interview.

Peter Collinson is a freelance consultant specialising in UNIX. He can be reached as pc@hillside.co.uk electronically (although your mailer might be happier to put the address the other way round) or by phone on 0227 761824.



STRETCH YOUR PASCAL AND FORTRAN PROGRAMS

Running out of memory is a common complaint these days. The first hurdle on most compilers is the 64K limit on code or data. All Prospero compilers have options to leap over this one. The next is the DOS 640K limit. With OS/2 this has never been a problem, and with the new Prospero DOS Extender Kit your DOS programs can now use all the memory on your machine.

The DOS Extender Kit can be used with Windows, DOS/16M and OS/286 and costs £120. It requires Prospero Pascal or Fortran, each costing £240.

Prospero Software
/ LANGUAGES FOR MICROCOMPUTER PROFESSIONALS
190 CASTELNAU, LONDON SW13 9DH, ENGLAND TEL: 081-741 8531 FAX: 081-748 9344

CIRCLE NO. 453

NOVELL Develops in **DATAFLEX**

The Benefits....

Buyers say...

"If I develop in DOS, I don't want to have to rewrite in OS/2" said Jim Baird, MIS manager for Novell Inc., in Provo, Utah. "DataFlex provides a good migratory path to OS/2 without losing what I have already developed in the DOS environment.

"With DataFlex under OS/2," he added, "I can put in one workstation and have a user do what would normally take two machines under DOS." With more than 140 users accessing as many as 200,000 records and having to maintain a response time of less than one second, Baird likes the product's performance.

"I haven't had any problems with the corruption of data," he said. "I have users running 10 different OS/2 windows who can be working in a spreadsheet and come back to find the reports."

Baird likes DataFlex OS/2 as a development environment because of its macro capability for quick and simple programs and its solid programming language for more experienced users.

"You could start writing in lower-level code if you wanted to develop a quick program, and you could have a prototype out in 10 or 15 minutes," he said.

DataFlex provides quick applications prototyping and development through its relational database features and solid 4GL programming language. Once developed, this application can be easily run on DOS Networks and OS/2, as well as AIX, Xenix, Unix and VAX/VMS. Thus protecting the user's investment.

Novell are not alone in their use of DataFlex for applications development. Central and Local Government departments, corporates and software developers are just some of those who have been choosing DataFlex since 1981 - creating a library of thousands of successful DataFlex applications.

DataFlex is a powerful applications environment geared to on-line transaction processing and providing rapid access to data for large numbers of users. DataFlex not only protects your application development through technology portability, but also through technology developments, with the new options of a Database Engine, distributed database, Object Oriented Programming, CUA interface and relational report writer.

Try it out for yourselves with our special introductory offer of an evaluation copy with full documentation at £99 (including VAT and carriage). Send back the coupon below or phone our sales desk on 071-729 4460.

PC. WEEK (U.S.A.)
February 19th, 1990

The performance database for applications development

Further information from your DataFlex dealer or:

**DataFlex (Information Management)
Services Limited**

**DataNet House, 114/116 Curtain Road,
London EC2A 3AH**

Telephone: 071-729 4460

Telex: 94018587

Fax: 071-739 1247

All Trademarks acknowledged

CIRCLE NO. 454

Please send me information about **DataFlex**. EV

Name _____

Company _____

Position in Company _____

Address _____

_____ Post Code _____

Phone _____ Date _____

What applications areas interest you? _____

**DataFlex (Information Management) Services
114/116 Curtain Road London EC2A 3AH**

EXE 291



C++ or C9X?

When you move to C++, do you expect to be able to recompile all your old C code straight off? Derek Jones explains why you cannot, and what to do about it.

Finally, we have an ANSI standard for C. You might well assume - and it certainly seems like it - that there has been a C standard since before the outbreak of World War II. The claims made by compiler vendors in their advertising certainly give this impression. In fact, the *Draft Proposed C Standard* lost its 'Draft Proposed' prefix on 14th December 1989, only just nipping in before the end of the decade. (*The decade begins in 1991* 'mob please don't write in to dispute this; your views are known - Ed.)

In this article, I do not intend to dwell on the current standard, but rather ask the question 'Where do we go from here?' The fact is that the recent emergence of C++ as a mainstream language has complicated the future of C.

C++ is widely viewed as the successor to C. Many programmers believe that once C++ has settled down (the language having just gone through a major revision), they can easily move their code from C to C++. However, things might not be that simple. Most

people assume, incorrectly, that C++ is a pure superset of C.

The C language standard has not been cast in concrete - the ANSI people are now thinking to the next revision

This view arises partly from the fact that many C++ compilers work by translating into C as an intermediate stage. The ability to be translated to C is no guarantee of compatibility, nor is it of any particular significance. Translators are available to con-

vert FORTRAN, COBOL, Pascal, BASIC, Ada, LISP and many other languages to C. Nobody would claim that these were supersets of C.

Thus the connection between C and C++ is not as strong as most C users probably think. Potential problems arise because the creators of C++ regarded it as a separate language, distinct from C, although they did aim for maximum compatibility with 'classic' C.

Who says that C++ is a different language from C? Three out of the four groups of people whose opinion signifies: (1) Bjarne Stroustrup, the originator of C++, (2) the ANSI C committee and (3) the majority of the current ANSI C++ committee. The fourth group is the C user community which, for the most part, is unaware of the situation.

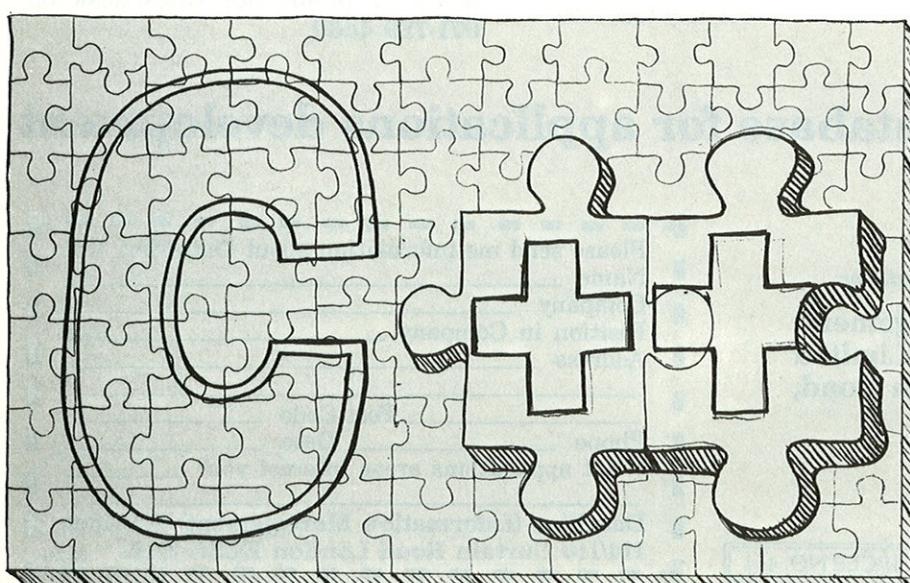
There is a further complication. The C language standard itself has not been cast in concrete. The people who worked on the ANSI C standard, X3J11, are now thinking ahead to the next revision of the language, C9X. The standards making rules require that all language standards be regularly reviewed and updated.

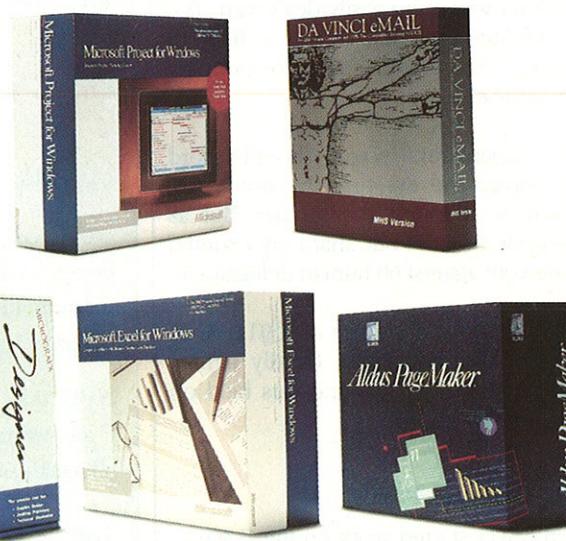
So what? Well, looking ahead five to 10 years, there is the a good possibility of ending up with two C-like languages which are mutually incompatible.

The question of C/C++ compatibility is currently being debated by the C++ committee. The main purpose of this article is to widen the audience of this discussion.

The beginning

In late 1988 ANSI (American National Standard Institute) received a proposal to set up a committee to standardise C++. Since a standards committee (X3J11) al-





Normally one plus one equals two. But when you add Microsoft C version 6.0 and the Microsoft Windows Software Development Kit version 3.0, things start to multiply.

As a matter of fact, these two Microsoft products were used to create every leading application for Windows on the market.

There's a good reason for this success. Because as the creators of the Windows environment, Microsoft understands the system and its potential for

personal computing. And that simply allows us to develop the best tools for creating applications for Windows.

So if you're serious about writing programs for the Windows environment, call your dealer and order Microsoft C and Windows SDK.

Together they equal everything you need to create the highest quality applications for Windows. Maybe even the next Windows sensation.

Microsoft®
MAKING IT ALL MAKE SENSE.

For more information, please call Microsoft on 0734 500 741. Microsoft Languages and Development Kits may be purchased from your usual dealer or direct from the following Microsoft Language Resellers: Grey Matter, 0364 53499. Software Construction Company, 0763 244114. System Science, 071 833 1022. QA Training, 0285 655888.
©1991 Microsoft Corporation - All rights reserved. Microsoft and the Microsoft logo are registered trademarks and Windows is a trademark of Microsoft Corporation. All other trademarks acknowledged.

ready existed for C, ANSI sent it a request asking what it thought should happen. X3J11 discussed two questions: 1) When should the C++ standardisation effort start? and 2) Should the work be done by the existing C committee?

One of the most vocal opponents against standardising C++ at once was the AT&T representative. He argued that C++ was still evolving, and that we should wait a few years before standardisation. Others pointed out that C++ was already being used for real work, and that early standardisation would reduce the number of language dialects that might have to be considered later. The voting was close, and the *Standardise now!* lobby won. The second point was more easily dealt with. The C committee regarded C++ as a different language, so the work should be done by somebody else.

(On a supernatural note, I should record that during the meeting the hotel was shaken by a minor earthquake. But as somebody pointed out, that only counted as one vote against 60 human delegates.)

Having seen the response of X3J11, ANSI set up a C++ committee (wittily named X3J16) and its first meeting was held on 15th December 1989.

The design aims

When X3J11 started work on the C standard, it specified its aims as preserving the spirit of C and codifying common existing practice. That is: standardise what exists, and don't add anything new. To a very large degree it achieved these aims.

What are the aims of the C++ committee? Nothing has been decided yet, but the base document from which they are working is *The Annotated C++ Reference Manual* by Ellis & Stroustrup. This book includes two language features described as experimental: templates and exception handling. Various groups are looking into these areas. But the hot topic of discussion at the moment concerns those parts of C++ inherited from C that people feel should be sorted out. (These exist because the C committee's policy of codifying existing practice resulted in some of the C's quirkier knots remaining unravelled.)

Uncle Sam's Standard

So the US went ahead and started the C++ standardisation process. This caused something of a fuss in the ISO (International Standards Organisation) which thought (and thinks) that the standardisation of C++

Maintaining C/C++ Compatibility

The information for many of the following differences was taken from a paper by Andrew Koenig and Bjarne Stroustrup. This list is not complete.

Identifier name spaces

C has different name spaces for object names and tag names. C++ has a single name space. Thus:

```
struct s_tag{int i,j;} s_tag;

enum count{ec1, ec2} x;

int count(void);

union u_info {int i; float f;};

typedef union u_info u_info;
```

are all legal C but illegal C++. The C programmer is relying on `s_tag` and `count` being in different name spaces. In the case of the `typedef` example, the C programmer is saving having to provide the keyword `union` when declaring an object of type `u_info`.

Syntax

C allows redundant parenthesis to appear in declarations, C++ does not. This only affects those programmers who use parenthesis as an aid to clarifying what is meant.

```
int (*obj);
int (f) (int);
```

are both legal C but illegal C++.

Keywords

C++ contains keywords that are not in C; `class`, `delete`, `handle`, `inline`, `new`, `operator`, `private`, `protected`, `public`, `template` and `virtual`. It is a simple matter to avoid using these reserved names as identifiers in C.

Type safe function calls

C++ requires that functions be declared before use. C has no such requirement - but it is good C programming practice anyway. The feature that will trip up most programmers is that the C++ declaration `extern int f()` declares a function that takes no parameters, ie the equivalent to `extern int f(void)` in C. Programmers that do not explicitly declare function prototypes are in for a lot of work.

```
extern void f();

main()
{
    f(1); /* Illegal C++, use differs from declaration */
    g(2); /* Illegal C++, not declared */
}
```

PC-lint presents C Bug # 571

```
#include <stdio.h>

main()
{
    char ch = 0xFF;
    if( (unsigned) ch == 0xFF )
        printf( "equal\n" );
}
```

What output does your C compiler produce and what warnings do you get? Is the output what you expected? If you can't find this bug, give us a call.

PC-lint will catch this and many other C bugs. Unlike your compiler, PC-lint looks across all modules of your application for bugs and inconsistencies.

"PC-lint, the one tool I believe *all* C programmers should have." - *The C Gazette*

More than 270 error messages. More than 90 options for complete customization. Suppress error messages, locally or globally, by symbol name, by message number, by filename, etc. **Easy to Use** - If you know C you already know how to use PC-lint. Check for **portability** problems. Alter size of scalars. Adjust format of error messages. Automatically generate ANSI prototypes for your K&R functions.

New Features in Version 4.0

Lint Object Modules - incremental linting
Weak Definition Checking - macros, structs, typedef's, enums, and declarations not used globally and locally. Header files not used. Externals that can be made static. Declarations that can be moved from headers.

Mainframe and Mini Programmers

Get **Flexelint** in shrouded source from for Unix, OS-9, VAX/VMS, QNX, IBM VM/MVS, etc. Requires only K&R C to compile but supports ANSI. Pricing starts at \$798. Call for details.

Gimpel Software

3207 Hogarth Lane, Collegeville PA 19426
 0101 (215) 584-4261 - CALL TODAY
 0101 (215) 584-4266 - FAX

PC-lint is only \$139 - Specify MS-DOS or OS/2

PC-lint 386 is only \$239 -
 requires MS-DOS on a 386.

For power users with huge programs.

30 Day Money-back Guarantee.
 PA add 6% sales tax.

PC-lint and Flexelint are trademarks of Gimpel Software.

CIRCLE NO. 456

Maintaining C/C++ Compatibility (continued)

C++ also enforces type checking across different translation units. C has no such requirement. Thus two variables declared with different types in different files will be flagged as an error at link time. Such occurrences are anyway likely to be bugs in the C code, and ought to be flagged.

An obscure problem arises for **struct** compatibility. C++ does cross translation unit checking by type name, while C uses structural comparison. Suppose file 1 contains

```
extern struct abc{int i, j;}x;
```

and file 2 has

```
struct xyz{int i,j;}x;
```

then this is legal C but illegal C++. This is because C++ only checks the tag names, while a C linker would check the ordering and type of the fields (although the Model Implementation is the only compiler/linker known to the author that performs this checking in C). Strictly speaking, the above example results in 'undefined behaviour', but most linkers will produce the hoped-for behaviour.

const

C++ treats objects declared with the **const** qualifier as static. C++'s **const i** is equivalent to **static const i** in C. In C, if you declare **const i** in two files, both refer to the same object. In C++, such a construct would refer to two separate objects, each local to the file in which it occurred.

To get around this potential problem, C programmers should precede declarations of **const** by **extern**, eg **extern const i** - not forgetting to give one of them an initialiser to create a defining occurrence.

Declarations

C allows multiple definitions of the same object in the same file at global level. This is not allowed in C++. The following is legal C but illegal C++:

```
char c;  
char c;
```

Such multiple declarations are usually the result of sloppy C programming, and should anyway be avoided.

Type of 'a'

In C, character literals are of type **int**, in C++ they have type **char**. This difference is only apparent when the type is required. **sizeof('a')** returns the size of an **int** in C and the size of a **char** in C++. Character literals are still converted to **int** the same way in both languages.

Array initialisation

C++ fixes a rather quirky C rule. In C, a string used to initialise an array of known size need not be zero terminated. So in

```
char x[2] = "ab";
```

the null character terminator is omitted. C++ insists that strings be null terminated, and the above construct is illegal.

should be done as an international effort - as Bjarne Stroustrup, along with Denmark, had originally proposed. The UK faction was also very keen for this to happen.

The C++ standard is now being driven by the US. Unhappily, the UK is currently without a national voice. IST/5 (the UK committee responsible for programming language standardisation) has decided not to set up a UK C++ panel, as a protest at the US action. Since C++ is growing in popularity, and may eventually rival C as the language of choice, IST/5's current position seems short-sighted. It is very unlikely to stop the US work, and prevents the UK from influencing the direction of C++.

What to do

Returning from the politics of Standards committees to the day-to-day problems of the programmer on the street, what can be done about the C/C++ incompatibility problem? I can see two approaches: 1) lean on the C++ committee, so that it makes C++ a pure superset of C and 2) write your code in a style that is compatible between C and C++.

Approach 2) I have described in a separate box. There are several ways of tackling

Approach 1) - none offer guaranteed success. You could write to the convener of the C++ ANSI committee (address supplied at the end of the article) contributing your

The current hot topic concerns those parts of C++ inherited from C that people feel should be sorted out

suggestion for the aims of the C++ standardisation committee.

You could adopt a 'wait and see' approach. Several years hence, the committee will announce the public review of the C++ draft standard. All proposed standards must go through this period of public review.

During these reviews (there are usually more than one, C went through three) copies of the draft standard may be purchased and readers may contribute their comments. These comments are collated and considered by the committee, which must give a response - although this can take the form 'Thank you for your comments, they have been considered but rejected'. The draft is updated in the light of these comments, and the cycle continues until no changes are made. However, by the time a draft standard is going through its public review, it is usually much too late to make fundamental changes.

C9X

The C standard itself is not fixed. The rules require that standards are reviewed every five years, although in practice the cycle usually takes a decade; work does not start for five years, then takes five years to complete. So we do have a C9X to look forward to.

Work has not yet formally started on the next revision of C. But two other activities have started. The ISO has given the go-ahead for the production of an Addendum to the ISO C standard, and X3J11 has set up an interpretations committee to handle queries on the meaning of the standard.

System Architect™

A PC based Microsoft Windows CASE tool

Microsoft Windows V 3.0 compatible

RAJAHIM SOFTWARE

118-120 Warwick Street,
Leamington Spa, Warks
CV32 4QY

TEL: 0926 450858

CIRCLE NO. 513

Deadlock-I

ONLY £7.50

1. Software protection units that are programmable by the Software House.
 2. No need to keep a stock.
 3. Ten (10) days delivery guaranteed.
 4. The codes can be changed if desired by the software house.
 5. Supplied with **FREE** software.
 6. Are you already using DONGLES? You can have us as a second source. (Call for details).

Please send all orders to B. L. Computer Security, 101 Hendon Lane, Finchley, London N3 3SH, or call us on 081-343 0734 or Fax us on 081-346 2672.

CIRCLE NO. 457

TOOLBOX WINTER '91 NOW AVAILABLE

EXE - DECEMBER 1991

THE SOFTWARE CONSTRUCTION CO. LTD.
1 THE MALTINGS, GREEN DRIFT, BOYSTON, HERTS, SG8 5DB (0763) 244114 FAX (0763) 244025

CIRCLE NO. 458

The UK representation, among others, was unhappy with the looseness of the drafting of the C standard. This document contains many undefined features, poorly defined constructs and ambiguities. So one aim of the ISO's addendum is to tighten up the language definition - without changing what it defines. (The list of problem areas is still being compiled, so if you have a copy of the Standard, and you think that you have found an omission or inconsistency, please contact me at the address given

below.) The addendum will also address a Danish request for an alternative, non-trigraph method of supporting characters such as [and { in the C source, and a Japanese request for run-time support of multi byte characters.

Many of the interpretation questions raised by the UK were addressed by ANSI X3J11 at its meeting in California during September last year. The volume of the UK questions swamped the meeting and the answering

of some of them had to be postponed. A surprising suggestion (in my view) was that at some future date, interpretation questions might be dealt with at ISO C meetings.

Finally, there is also interest in extending the C standard in specific areas. A numerics group has already been set up, and several meetings have been held. If you are interested in this, please contact Rex Jaeschke at the address below.

Will X3J11 eventually start work on the next version of the C standard, or will it yield to C++? We will have to wait and see.

EXE

Derek Jones is a member of the UK C Panel and has represented the UK at ISO C meetings. He is also the technical editor of the UK component to the ISO C Addendum and the project editor of the ISO C Addendum.

The UK C panel meets every three months in central London. If you would like to take part, please contact Cornelia Boldyreff (address given in box) for details. Contact Neil Martin at the BSI for details of C++ standardisation.

Some useful addresses

UK C++ standardisation

Neil Martin
BSI Quality Assurance
PO Box 375
Milton Keynes MK14 6LE
email: neil@bsiqa.co.uk

UK C Panel Convener

Cornelia Boldyreff
Department of Computer Science
Brunel University
Uxbridge
Middlesex UB8 3PH
email: Cornelia.Boldyreff@brunel.ac.uk

Editor ISO C Addendum

Derek Jones
Knowledge Software Ltd
32 Cove Road

Farnborough
Hants GU14 0EN
email: derek@knosof.co.uk

ANSI C++ Convener

Dmitry Lenkov
California Language Laboratory
Hewlett Packard
19447 Pruneridge Avenue
Cupertino CA 95015
USA
email: dmitry%hpda@hplabs.hp.com

Convener C Numerics group

Rex Jaeschke
2051 Swans Neck Way
Reston VA 22091
USA
email: uunetaussie!rex

CLIPPER ADD-ON SOFTWARE

QBS Software Limited specialises in providing the best add-on libraries and utilities for Clipper. We distribute throughout Europe providing local support centres in France, Germany, Italy and elsewhere through our agents.

FUNCky	General Function Library
Blinker	Dynamic Overlay Linker
Netlib	Networking Library
SilverComm	Communications Library
SilverPaint	Graphics Library
SubNtx()	Filtering Utility
SpellCode	Spell Checker
The Engine	Linkable Spell Checker
Biton	Oracle Library
Scripton	Postscript Library
Overlay()	Memory Roll Out Utility

For further information, free demo software, prices and how to order please contact **QBS Software Limited**,
10 Barley Mow Passage, London W4 4PH
Tel: 081-994 4842, Fax: 081-94-3441
BBS: 081-747-1979

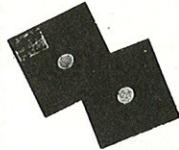
All trademarks recognised.

CIRCLE NO. 459

BRENT

Communication and Telemetry Consultants

"MaxPro - the ultimate in hardware for software protection"



The MaxPro Data key fitted to the Parallel Port is totally transparent to the end user.

Software customers make as many backups as are required but of course only the machine using the individually coded MaxPro key will run the program.

BEWARE THE PIRATE'S PATCH

High security encryption is created by the Simple-to-Use Menu-driven MaxPro attachment software. Other data keys may be patched out in the software rendering it totally vulnerable. MAXPRO protected files cannot be patched. Choose whether or not to shut down the program if tampering occurs.

Optional limited life and Usage information collection facilities are provided.

No source or object code modules needed.

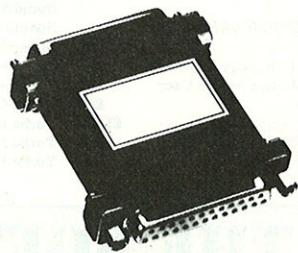
The MaxPro system works on IBM; PC; XT; AT Compatibles and PS2.

For additional information contact us at:

Brent Communications

Unit 2
Dragon Industrial Estate
Harrogate HG1 5DN

Tel: (0423) 566972
Fax: (0423) 501442



CIRCLE NO. 460

DAILY NON-STOP DELIVERIES TO EUROPE!

AT UNBEATABLE PRICES!

**PRE-SALE
CONSULTING**

**POST SALE
SUPPORT**

NEW AND NOTEWORTHY:

Calera's WordScan 286/386	£ 364
Probably the best implementation of optical character recognition today.	
Expanz Disk Expander Card	£ 119
Triple hard disk storage without performance loss, make your floppies run 4 times faster	
GEOWORKS Ensemble	£ 99
Multitasking, virtual memory, Motif user interface available to ANY PC, including an XT.	
DESQview/X	CALL
Brings the complete X Window System to DOS.	
Provides distributed graphics and remote computing capabilities.	

ARTIFICIAL INTELLIGENCE:

Arity Combination Pack v5.1	£ 660
BrainMaker	£ 149
BrainMaker Professional	£ 639
PDC ProLOG DOS v.5.21	£ 190
PROLOG-2 Personal	£ 90

ASSEMBLERS & LINKERS:

Blinker v1.1	£ 145
Microsoft Assembler v5.1	£ 88
Plink/LTO v2.3	£ 300
RT-Link Plus v4.1	£ 300
PolyLibrarian II	£ 92
Spontaneous Assembly	£ 152

BACKUP:

Back-It	£ 56
Copy II PC v6.0	£ 25
Copy II PC Deluxe Option Board	£ 105
Central Point Backup	£ 55
Norton Backup	£ 89

BASIC:

HyperHelp	£ 82
Microsoft BASIC PDS v7.1	£ 287
Microsoft Quick Basic v4.5	£ 60
PowerBASIC v2.1	£ 69
ProBas	£ 130

BUSINESS

Ami v1.2	£ 112
FormFiller v3.0	£ 79
FormTool Gold v3.0	£ 54
Grammatic IV	£ 47
GrandView v2.0	£ 177
LetterPerfect	£ 119
Lotus 1-2-3 v3.1	£ 382
Lotus Freelance Maps US & Int'l	£ 244
Mace Vaccine	£ 55
Microsoft Office for Windows	£ 580
Norton AntiVirus	£ 78
ORG Plus Advanced v5.0	£ 59
Quattro Pro v2.0	£ 277
Sidekick v2.0	CALL
Sideways v3.3	£ 53
SQZ! Plus	£ 62
Super Project Expert v2.0	£ 394
TimeLine v4.0	£ 413
WordPerfect v5.1	£ 240

C:

ASYNCH MANAGER v3.0	£ 116
C-terp w/Source v3.3	£ 182
C-Worthy w/Forms & Source	£ 349
Clear + for C	£ 115
Control Pak/Windows	£ 360
Essential Communications v3.0	£ 184
FairCom Toolbox Spec'AirCom	£ 420
Greenleaf CommLib v3.0	£ 236
Lattice C Dev Sys for DOS & OS/2	£ 153
Microsoft C Compiler v6.0	£ 287
Microsoft Quick C/Quick Asm. v2.51	£ 117
PC-Lint	£ 73
Turbo C Tools	£ 92
WATCOM C /386 Professional	£ 845

CADD:

3-D Drafting v1.1	£ 188
AutoCAD Release 11	£ 2656
Design CAD 3D v3.1	£ 183
Generic CADD v5.0	£ 212

CASE:

CASE: Windows	£ 839
CASE: PM for C	£ 1679
CASE: PM for C++	£ 1679
Competitor II	£ 213
EasyCASE Plus	£ 239
Teamwork/PCSA	£ 898

COBOL:

Microsoft COBOL v4.0	£ 520
LPI COBOL	£ 975
RM/COBOL-85 v5.0	£ 707

COMMUNICATIONS:

Carbon Copy Plus + Host	£ 103
CO/Sessions Complete	£ 152
CrossTalk Mk. 4 v2.0	£ 125
LapLink III	£ 78
Mirror III v2.0	£ 77
PC Anywhere IV (Host & Remote)	£ 92
Procomm Plus v1.1b	£ 62
Remote2 (Host & Caller)	£ 100
SmartTerm 240 v3.1	£ 175
Takeover	£ 150

DATABASE & FILE MANAGEMENT:

4GL Rapid Dvlp System w/Debugger	£ 671
ALPHA Four v1.1	£ 277
Btrieve for DOS	£ 360
Clarion Professional Developer v2.1	£ 425
Clipper v5.0	£ 470
DB FILE & DB_Retrieve w/Object Srcx	£ 209
dBASE IV v1.1	£ 448
dBFast/Plus v1.4	£ 158
DBXL Diamond v1.3 R	£ 127
FairCom Sql Server	£ 297
FoxPro v2.1	£ 181
FoxPro	£ 416
Gupta Sql Windows v1.3	£ 795
Novell XQL Relational Data Mgr	£ 496
Paradox 3.0 Competitive Upgrade	£ 129
Paradox v3.5	£ 469
Paradox SQL Link	£ 325
Professional Oracle Developers Kit	£ 609
R&R Clipper/Foxbase+ Module	£ 33
R&R Report Writer for XDB SQL	£ 182
R:BASE v3.1	£ 440

LAN & LAN-READY SOFTWARE:

FoxPro Multiuser	£ 572
Framework III 5 (Users)	£ 568
LANtastic Ethernet Adapter	£ 227
LANtastic Ethernet Starter Kit	£ 469
Microsoft Windows Add'l Station	£ 75
Microsoft Word Add'l Station	£ 172
Networker Plus	£ 132
Novell ELS I Netware v2.12	£ 496
Novell ELS II Netware v2.15c	£ 1178
Novell: Entire product line	CALL
Paradox Network Pack v3.5	£ 590
Q:DOS II (8 User)	£ 100
R:BASE v3.1 (5-User) LAN Pack	£ 560
Saber LAN Administration Pack v1.1	£ 329
Saber Menu for Windows v3.0	£ 234
Super Project Expert (3-User)	£ 394
TimeLine v4.0 1 Add'l user	£ 178
Ventura Publisher Gold Add'l Node	£ 263
WordPerfect Add'l Station	£ 150
StationServer v6.0	£ 78
XtreeNet v2.0 for ELS Level 1 & 2	£ 142

PASCAL:

ASYNCH PLUS v5.02	£ 113
B-Tree Filer	£ 78
DESQview API Toolkit	£ 250
Flash-up v3.05	£ 56
Microsoft Pascal v4.0	£ 173
Object Professional v1.1	£ 116
Pascal ASYNCH MANAGER v1.1	£ 105
Pascal TOOLS & TOOLS 2	£ 108
POWER SCREEN	£ 92
POWER TOOLS PLUS v5.1	£ 89
Turbo Analyst w/source	£ 80
Turbo Pascal v6.0	£ 90
Turbo Pascal Professional v6.0	£ 179
Turbo Plus	£ 132
Turbo Professional	£ 80
View232 v1.0	£ 113

SCIENCE & ENGINEERING:

Advanced Math Applications Pack	£ 60
Derive v1.6	£ 139
MathCAD v2.5	£ 285
Mathematica/386 v1.2	£ 559
SSP/PC Scientific Subroutine Library	£ 230

SCREEN DESIGN & WINDOWING:	
Flash-Up Developer's Toolbox	£ 51
Greenleaf DataWindows w/Source	£ 300
PANEL Plus II	£ 253
Vermont Views v2.0	£ 300

TRANSLATORS:

dBx/dBprt v4.7	£ 753
FOR_C w/Binary Runtime	£ 408
FOR_C++	£ 813

UNIX/XENIX PRODUCTS:

4GL RDS/ID for SCO Unix	£ 2077
Grammatic IV	£ 180
Interactive 386/ix Motif Window Manager & Development System v1.2	£ 272
Microsoft FORTRAN Compiler	£ 434
Norton Utilities for System V	£ 180
Oracle Unix/Xenix	£ 1378
Pascal Compiler for XENIX/286	£ 434
Quicksilver/UNIX 386 SCO	£ 500
RM/COBOL-85 for UNIX/XENIX 386 (4-User)	£ 1163
SCO Portfolio Suite 386	£ 992
VEDIT PLUS for Unix/Xenix	£ 174

WINDOWS 3.0 APPLICATIONS:

Actor for Windows w/Lang. Exts	£ 665
Adobe Illustrator v1.1	£ 255
Adobe Type Manager	£ 53
Btrieve for Windows	£ 372
Charisma v2.0	£ 360
CrossTalk for Windows v1.1	£ 113
Turbo C++ Competitive Upgrade	£ 133
Turbo C++ Professional	£ 177
Zortech C++ Compiler v2.1	£ 123
Zortech C++ Developer's Edition v2.1	£ 273

OPERATING ENVIRONMENTS:

Asymetrix ToolBook	£ 263
Brief Editor	£ 170
Btrieve	£ 360
Control Pak/PM	£ 360
Describe Word Publisher v2.0	£ 366
GSS Graphics Devel. Toolkit	£ 520
IBM DisplayWrite 5/2	£ 221

OS/2 PRODUCTS:

Asymetrix ToolBook	£ 263
Brief Editor	£ 170
Btrieve	£ 360
Control Pak/PM	£ 360
Describe Word Publisher v2.0	£ 366
GSS Graphics Devel. Toolkit	£ 520
IBM DisplayWrite 5/2	£ 221

We will match any advertised Price!

- Prices are ALL-INCLUSIVE and include 1st Class Recorded Delivery.
- All major credit cards accepted.
- 30 day invoicing for qualified organizations.
- Special discounts for large or corporate buyers.
- Prices subject to change without notice.

Here are the Bugs - TopSpeed C

Last year we printed a letter of appeal from the Romanian programmer Doru Turturea. JPI kindly responded by sending a copy of its TopSpeed C compiler. Here's how Doru, together with his friend Dan Somnea, repaid the kindness!

A few months ago, by courtesy of Jensen & Partners International UK Ltd and the Editor of .EXE Magazine, we received a copy of the TopSpeed C compiler set, consisting of twelve 360 KB disks and various manuals. It so happened that we had read the *Here are the Bugs* articles by Mr Neil Martin published in .EXE Magazine October and December 1989. We decided to attempt something similar with our own compiler - not biting the hand that fed us, just giving it a friendly nip.

**We'd expected the
%o specifier to
take the first two
digits from the
"129e-2" literal**

For this paper, we tested V1.04b of the TopSpeed C compiler with V2.02 for the Integrated Development Environment (IDE).

First of all, we tried out Mr Neil Martin's examples of bugs found in Microsoft C5.1 and Turbo C V2.0 implementations. With one exception, the TopSpeed C Compiler passed all its exams. In the case of Bug #6 presented in that paper - see below for a listing - the TopSpeed C syntax analyser did not detect the 'too large' value

```
#include <stdio.h>
main()
{
    /* Declare test variables of */
    /* all the standard types      */

    char c;
    int i,j,k;
    float f;
    int p;
    char s[10];

    /* Carry out sscanf tests */
    sscanf("129e-2", "%c", &c);
    sscanf("129e-2", "%o%d%x", &i, &j, &k);
    sscanf("129e-2", "%e", &f);
    sscanf("129e-2", "%p", &p);
    sscanf("129e-2", "%s", &s[0]);

    /* Print results of test */
    printf("c = %c\n", c);
```

```
printf("i j and k = %x %x %x \n",
       i, j, k);
printf("i = %x \n", i);
printf("p = %x \n", p);
printf("string = %s \n", s);
```

TEST RESULTS

When run, the programs output was as follows:

c = 1	good!
i j and k = 2 9 e	instead of 'a 9 e'
f = 1.290000	good!
i = 2	good!
p = 129e	good!
string = 129e-2	good too!

Figure 1 - The sscanf() function

Affordable CASE Tools

Personal-SELECT at £495

A full function PC-based Analysis and Design Tool supporting the Yourdon, Ward-Mellor, Hatley or HOOD methodologies.

Project-SELECT from £995

Links Personal-SELECT users working in projects and teams in a true multi-user environment.

The SELECT product range is developed and supported in the UK by SELECT Software Tools. It is marketed directly and through a variety of third party distribution channels.

Call SELECT Software Tools today to order your personal copy or to request further information.



SELECT Software Tools Limited
Ullenwood Court, Ullenwood, Cheltenham,
Gloucestershire GL53 9QS

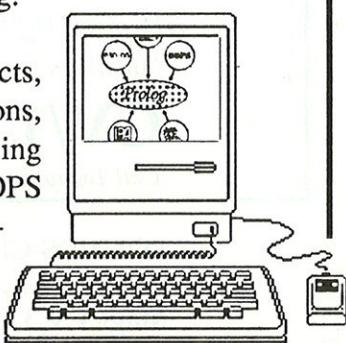
Telephone: 0242 226553 Fax: 0242 251491

CIRCLE NO. 462

OOPS + AI -> Prolog⁺⁺

Now you can combine the best features of AI and OOPS with LPA's new Prolog⁺⁺, the all new object-oriented extension to Prolog.

Prolog⁺⁺ integrates objects, methods, messages, functions, attributes - in short, everything you would expect in an OOPS environment - with the declarative power of Prolog.



All of the standard features of LPA Prolog, such as source level debugging, incremental compilation, sophisticated graphics and windows, and so on, are inherited by Prolog⁺⁺.

So why not be objective, and give us a call?



Logic Programming Associates Ltd
Studio 4, RVPB, Trinity Road, London,
SW18 3SX, England

Phone: 081 871 2016
Fax: 081 874 0449

386 Version
now available

CIRCLE NO. 463

System Science

C COMPILERS

Microsoft C 6.0 with PWB	£245.00	Introl C or Modula-2	£1445.00
Quick C & Quick Asm 2.51	£99.00	Aztec C Cross	£275.00
Quick Cver 2.5	£65.00	IAR C Cross with CSpy	£call
Turbo C++ Professional	£199.00	2500AD Cross AsM	£135.00
Zortech C++ Dev Ed	£275.00	Simulators & others	£call
Watcom C 8.0	£255.00		
Aztec C86 Comm	£265.00		
C compilers for 386	£call		

CROSS DEVELOPMENT

Aztec C Cross	£275.00
IAR C Cross with CSpy	£call
2500AD Cross AsM	£135.00
Simulators & others	£call

FORTRAN

MS Fortran-77 5.0	£235.00
Lahey F77L-EM/32 & 386 OS	£795.00
Salford F77/386	£765.00
Lahey Personal Fortran	£79.00
InGraf for Fortran (source)	£175.00

GRAPHICS & DATAFILE

Halo	£195.00
GSS XVT for Windows	£345.00
Metawindows	£175.00
Essential Graphics (no roy)	£275.00
PCX Prog Toolkit (new 4.0)	£165.00

C DATA & MISC

Btrieve for DOS	£345.00
XQL for Btrieve	£475.00
Ctree (source)	£195.00
CodeBase 4.2 (source)	£175.00
db_VISTA III	£545.00
GL_Financial Mathlib	£245.00
Blaise C Tools Plus	£89.00

C SCREENS & COMMS

Cscape, Look & Feel	£370.00
Vermont Views & Designer	£295.00
Panel Plus II	£275.00
Aspen Formation (new ver)	£195.00
C Worthy & cwARCHITECT	£call
Blaise C Aynch Manager	£115.00
Greenleaf CommLib	£185.00

PROGRAMMERS TOOLS

MKS Toolkit (Unix toolset)	£155.00
MKS RCS (Rev Cont Sys)	£125.00
PVCS Prof (lower price)	£375.00
PolyMake	£135.00
vsDesigner (Case Tool)	£475.00
Brief Editor 3.0	£195.00
Kedit Editor	£110.00
Flow Charting 3	£175.00
Dan Bricklin DEMO II	£195.00
MS Progs Lib 1.2 CD	£225.00
Personal Rexx	£110.00

OTHER LANGUAGES

Smalltalk V	£69.00
TopSpeed Modula-2	£145.00
MS Cobol 4.0 (new ver)	£445.00
RM Cobol-85 ver 5.0	£995.00
LM1 or Forth	£225.00
PC Logo	£50.00
Clipper 5.0	£395.00

MEMORY, MATHS etc

386 Max 5.0	£75.00
QEMM/386 5.1	£60.00
QRAM for 286	£55.00
Above Disc (LIM Emul)	£69.00
Norton Utilities (new ver 5)	£110.00
Spin Rite II	£69.00
Derive (Symbol Alg) 2.0	£135.00
MicroStat II	£295.00
Mathematica by Wolfram	£call

WINDOWS 3.0

MS Windows 3.0	£85.00
MS Windows 3.0 & Mouse	£125.00
MS Windows SDK 3.0	£325.00
Actor 3.0 with Lang Ext	£625.00
WinPro/3 from Xian	£525.00
WindowsMaker	£595.00
Source & Bios (DisAsm)	£125.00
Toolbook by Asymetrix	£295.00

"OUR RANGE INCLUDES MANY MORE ITEMS FOR DOS, OS/2, UNIX ETC. PLEASE ENQUIRE"

3-5 Cynthia Street,
London N1 9LF
Fax: 071-837 6411

Prices are subject to change and alteration Prices include UK Delivery
but are exclusive of VAT



071-833 1022

CIRCLE NO. 463

CIRCLE NO. 464

POWER DEBUGGING

BOUNDS-CHECKER

Finds out-of-bounds memory accesses — AUTOMATICALLY.

Flush out those Nasty pointer problems and other out-of-bounds memory accesses — AUTOMATICALLY.

Each time you make a change to a program, run BOUNDS-CHECKER while testing the new code. If you accidentally access out-of-bounds memory, BOUNDS-CHECKER will pop up displaying the offending SOURCE LINE. And your program runs at full speed.

Ship Bug-Free Products

You can run BOUNDS-CHECKER while testing your program. There are no additional steps to your testing cycle, but you can feel secure when the program has passed through BOUNDS-CHECKER with no reported problems.

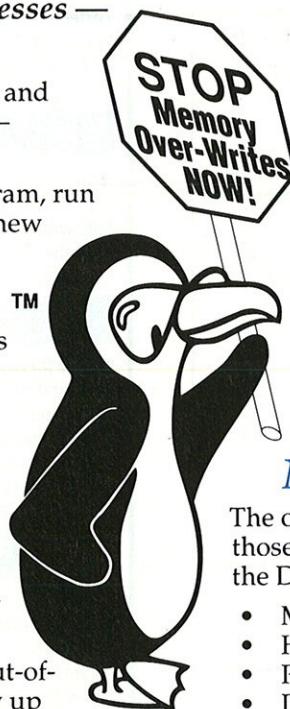
Many over-write problems and other out-of-bounds memory accesses do NOT show up during normal testing. An out-of-bounds memory location may be modified, but that particular location doesn't happen to be important at the time. Once the program is in the field and a certain network is loaded or a certain T&SR or device driver is loaded, that memory location suddenly becomes very important... AND THE SYSTEM CRASHES.

You can prevent these problems by making BOUNDS-CHECKER a standard part of your testing procedure.

Gives you the protection of a protected operating system under MS-DOS.

BOUNDS-CHECKER uses the 386 virtual machine technology to provide real-time memory protection. In addition BOUNDS-CHECKER uses the symbolic information output by your compiler to differentiate CODE and DATA. When your program is running, BOUNDS-CHECKER protects the program's CODE and all memory outside your program.

Requires 386 PC.
MS-DOS is a trademark of Microsoft Corporation.
386 is a registered trademark of Intel Corp.



"BOUNDS-CHECKER and Soft-ICE make sophisticated use of the most powerful versions of Intel's processor family to track down some of DOS programming's most insidious bugs. If you're developing programs for DOS, these are essential tools."

PC Magazine
July, 1990 pg. 48

Soft-ICE 2.5 New Version, New Features

The only debugger specifically designed to solve those problems unique to MS-DOS that we call the DOS Nasties.

- Memory over-writes
- Hung programs
- Program too big to debug
- Debugging T&SRs and Loadable Drivers
- Multiple Symbol Tables
- Supports Microsoft C 6.0 & Turbo C++

Windows Developers

Run CODEVIEW for Windows on a Single Monitor

Nu-Mega has done it again!

CV/1™ \$129

Call Today, Have it Tomorrow

BOUNDS-CHECKER	\$249
Soft-ICE 2.5	\$386
MagicCV 3.0	\$199
CV/1™	\$129

Special Offer...

Buy BC & S-ICE	Save \$100
Buy S-ICE & MCV	Save \$86
Buy all three	Save \$186

International Shipping charge is \$40.00
Credit Cards or Wire Transfers only
30 Day Money-Back Guarantee



CALL TODAY 0101 603 888-2386 or FAX 0101 603 888-2465
P.O. BOX 7780 ■ NASHUA, NH ■ 03060-7780 ■ U.S.A.



of the expression `LONG_MAX * 8` - but we suppose that this is really the job of a *lint*.

```
#include <limits.h>
main()
{
    int j = 0;
    switch (j)
    {
        /* TopSpeed allows */
        /* over large value */
        case LONG_MAX * 8 :
            j = 1;
    }
}
```

The next question was: how close is the TopSpeed compiler to the ANSI C Standard? Taking Appendix A of the K&R book *The C Programming Language* (2nd edition) as our guide, we discovered the following bug. We presume that the last changes in the ANSI Standard are not relevant to this particular bug.

The `sscanf()` function

The Bug. In Figure 1, the `%o` specifier seems to be treated incorrectly. We'd expected the `%o` specifier to take the first two

digits from the "129e-2" literal. In this case, 12 octal is 10 decimal or 0A in hexadecimal code. The observed result was 2,

The TopSpeed C package constitutes a very serious reply to other IDEs

which is wrong. The other two specifiers `%d` and `%x` were treated correctly.

The Fix. Avoid the `%o` specifier.

Conclusion

Despite these small problems, the TopSpeed C package constitutes a very serious reply to other IDEs. Its main trumps are:

- the editor has nine windows

- the error messages are more friendly
- you can use all the tremendous graphics primitives of Turbo C
- the MS-DOS DLL mechanism makes it closer to OS/2
- there is a powerful set of utilities subordinated to the same IDE

and so on.

We intend to continue its testing. If we discover any other relevant faults, we shall send them to you immediately!

EXE

Mr Dan Somnea is Senior Analyst and Assistant Professor in the Cybernetics, Informatics and Statistic Department. He is a member of the Romanian Computer Science Society.

Mr Doru Turturea is researcher in the Institute of Computing Techniques. He is the secretary of the Romanian Computer Science Society.

*Both the authors can be contacted at :
ECON Cybernetics and Informatics Dept,
Bucharest, 15-17, Calea Dorobanti,
Sector 1 71131, ROMANIA*

**GENUS
SYSTEMS**

3L Transputer Products

THE industry-standard range of Transputer compilers and support tools.

Languages available are:

Parallel C:

K&R with many ANSI extensions including function prototyping.

Parallel Pascal:

Standard ISO Pascal including conformant arrays with many useful extensions controlled by a compiler switch.

Parallel Fortran:

Full ANSI Fortran-77 with many extensions.

Tbug:

Debugger capable of observing and interacting with one or more concurrent tasks and many threads of execution.

Genus supplies these products both for MS-DOS and for QNX using our proprietary TransputerNet interface.

Language prices from £750

QNX

THE truly distributed high-performance operating system for PC and PS/2 class computers.

For wide-class of applications it offers facilities, performance and expandability unmatched at any price. Features include:

- Process-based with peer-to-peer messaging providing uniform access to all system resources network-wide.
- Name server supporting dynamic configuration of distributed applications.
- 1 to 255 computers per system.
- Protect-mode operation.
- Compact with real-time quality response.
- High performance file system with caching disc drivers.

Contact us now for prices and reasons why around 100,000 others world-wide have already invested in QNX.

System prices from £395

Further info, demo disks etc from:-

**Genus Systems Ltd, 9 Waters Close, Leith, Edinburgh EH6 6RB
Tel: 031-554 0455 Fax: 031-554 0456**

SEE US AT CeBit 91, HANOVER. HALL 4, STAND E48F47

Name Mangling in Turbo C++

C++ achieves its type-safe linking with a technique called 'name-mangling'.

*Borland's Alex Lane explains how Turbo C++'s implementation works
and why it differs from the cfront approach.*

Many newcomers to C++ are unaware that C++ compilers do name mangling. As they advance, however, users begin to deal with mangled names and invariably express an interest in the scheme used to encode names. This article presents the mangling scheme used in Turbo C++ and briefly touches on its advantages for C++ programming.

Background

Although support for object-oriented programming is perhaps the most significant feature of C++ that distinguishes it from C, there are nonetheless several other ways in which C++ enhances C.

One of the features of C++ that help make it a better C is type-safe linkage which fits in with the C++ philosophy of making life easier for the programmer.

All declarations and all uses of all variables, functions and other identifiers in C++ must adhere to type rules. However, enforcing this consistency is not a job for the compiler because only part of the code is compiled at any given time. The linker, on the other hand, handles all of a program's code (at least in object code form) and is best equipped to do type checking. There are problems associated with modifying linkers, however.

First, such changes would be substantial, since linkers currently have only a limited understanding of type; second, should we actually make the modifications, users would have to deal with multiple linkers on their systems. The best course is to take advantage of the capabilities of existing linkers to deal with names embedded in object files; C++ does this by embedding the associated type as part of the name. This is where name encoding, or mangling as it's commonly called, comes into play.

Turbo C++ name mangling

There are four basic forms of encoded names in Turbo C++:

1. @className@functionName\$args

This encoding denotes a member function `functionName` belonging to class `className` and having arguments `args`. Class names are simply encoded directly, eg a class `className` will appear as:

`@className@...`

in an encoded name. The encoding of function names and arguments is discussed below.

2. @functionName\$args

This form of encoding denotes a function `functionName` with arguments `args`.

3. @className@dataMember

This form of encoding denotes a static data member `dataMember` belonging to class `className`. Names of classes and data members are encoded directly eg a member `myMember` in class `myClass`:

`@myClass@myMember`

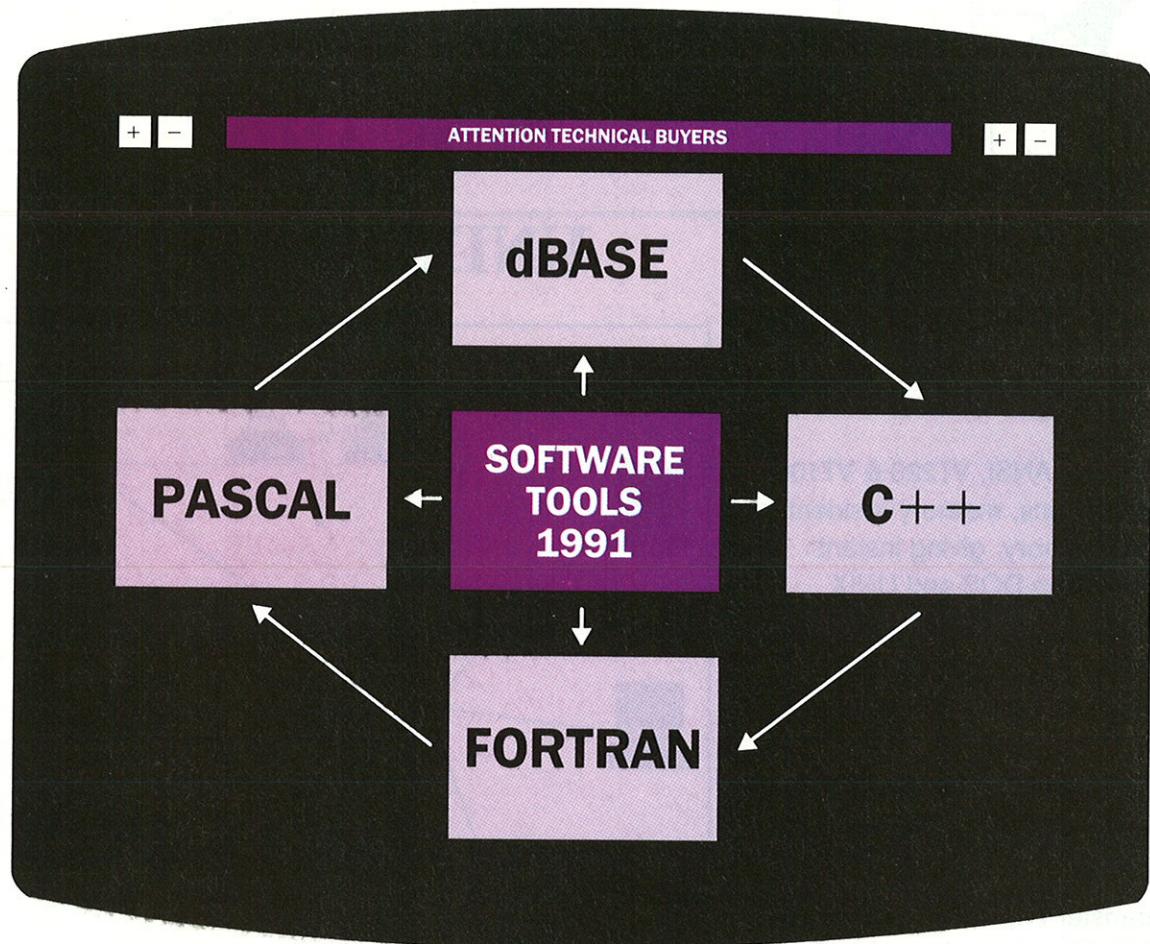
4. @className@

This name denotes a virtual table for a class `classname`. As noted previously, class names are encoded directly.

Character sequence	Meaning	Character sequence	Meaning
ctr	constructor	lsh	<<
dtr	destructor	lss	<
add	+	mod	%
adr	&	mul	*
and	&	neq	!=
arow	->	new	new
arwm	->*	not	!
asg	=	or	
call	()	rand	&=
cmp	~	rdiv	/=
coma	,	rish	<<=
dec	--	rmin	--
dele	delete	rmod	%=
div	/	rmul	*=
eql	==	ror	=
geq	>=	rplu	+=
gtr	>	rrsh	>>=
inc	++	rsh	>>
ind	*	rxor	^=
land	&&	sub	-
lor		subs	[]
leq	<=	xor	^

Figure 1 - Encoding for constructors, destructors and overloaded operators

Meet people who speak your language



Whatever your specialist interest you'll meet someone at Software Tools '91 who shares it. Because at Europe's largest and best event for software development you'll find more than just sales talk.

You'll meet people who really understand their products AND your requirements. And you'll have the chance to roll up your sleeves and get some real 'hands on' experience on the latest products. All in all, the perfect environment to find real solutions.

Special features at this year's show include:

- A PC Development Area
- A People and Skills Centre
- A Technical Preview Centre
- An exciting line-up of competitions and awards

All this, and a conference which includes some of the best known experts in the field talking about all that's current – and controversial – in software development today.

For more info plus your free tickets to Software Tools '91 just return this coupon to Lynne Davey, Blenheim Online, Blenheim House, Ash Hill Drive, Pinner Middx HA5 2AE, before 3rd May 1991.

11-13 June 1991
Wembley Exhibition
Centre, London.

CIRCLE NO. 467

Sponsored by

Software MANAGEMENT

Supported by

EXE



Name _____

Initials _____ Mr/Mrs _____

Company _____

Job Title _____

Address _____

Tel _____ Fax _____

I am interested in exhibiting at Software Tools '91.
 I am interested in visiting Software Tools '91.
 I would like information on the Conference.

Blenheim Online, Blenheim House,
Ash Hill Drive, Pinner, Middlesex
HA5 2AE, UK,
Telephone: 081-868 4466, Fax: 081-868 9933,
Telex: 923498 ONLINE G.

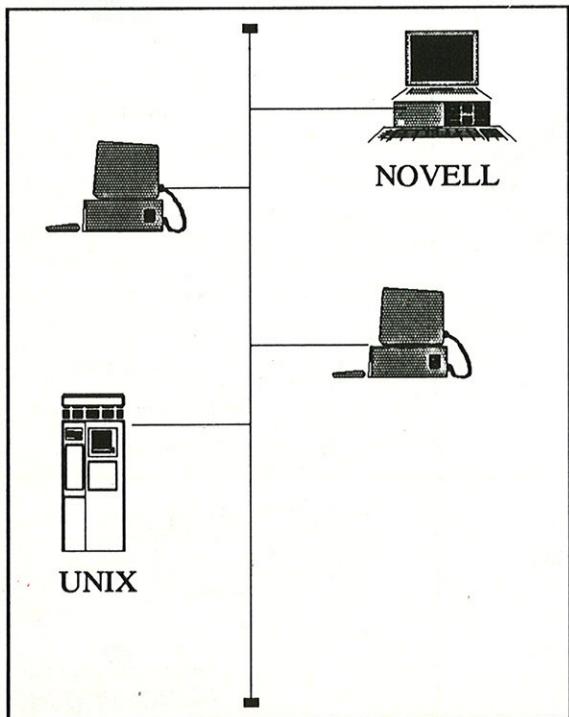
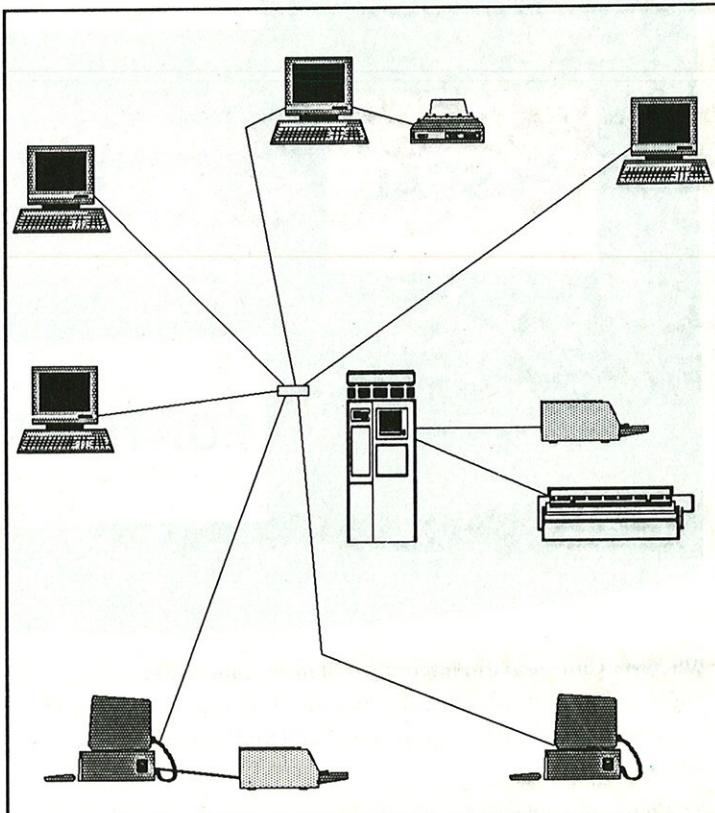

BLEHEIM ONLINE
The IT event specialists



SMOOTH SKATING BETWEEN DOS AND UNIX/XENIX

*** ICE.TEN ***

- Exact WYSE 60, ANSI, VT220 & VT100 terminal emulations, memory resident in 42K of DOS memory, giving instant switching between DOS and UNIX screens.
- Uses standard RS232 ports/cables
- Support for ANSI colour codes in all emulations.
- Fully programmable function keys
- Transparent and local print to PC's parallel port.
- Easy file transfer between PC and Host with an MSDOS style copy command.
- DOS applications can share UNIX printers.
- From £350 for an UNLIMITED no. of PC's linked to a single host.



*** ICE.TCP ***

- A DOS version of TCP/IP with the same exact terminal emulations as ICE.TEN
- Fast and simple file transfer.
- Can coexist with Novell and other DOS networks
- Drivers for most popular Ethernet cards.
- £195 for 1 PC, £995 for 10 PC's, £2995 for 25 PC's, £4995 for 100 PC's.

Custom Business Systems Ltd
TEL: 071 323 2297
MSDOS to UNIX experts.

Encoding functions

The encoded functionName may denote either a function name, a function such as a constructor, destructor or overloaded operator or a type conversion.

Ordinary function names are encoded directly eg:

`foo(int) → @foo$qi`

`sna::foo(void) → @sna@foo$qv`

The string '\$qi' denotes the integer argument of function `foo`; '\$qv' denotes no arguments in `sna::foo`. Argument encoding is explained in detail below.

Constructors, destructors and overloaded operators are encoded with a '\$b' character sequence, followed by a character sequence from the table given in Figure 1. So, for example,

Character sequence	Meaning
up	huge
ur	_seg
u	unsigned
z	signed
x	const
w	volatile

Figure 2 - Encoding of type qualifiers

Character sequence	Meaning
v	void
c	char
s	short
i	int
l	long
f	float
d	double
g	long double
e	...

Figure 3 - Encoding of built-in types

Character sequence	Meaning
<a digit>	(an enumeration or class name)
p	near *
r	near &
m	far &
n	far *
a	array

Figure 4 - Encoding of non built-in types

`operator+(int) → @$badd$qi`

`plot::plot() → @plot@$bctr$qv`

`plot::~plot() → @plot@$bdtr$qv`
The string '\$qv' denotes no arguments in the `plot` constructor or destructor.

`foo::myfunc`
(`myClass` near&) →
`@foo@myfunc$qr7myClass`

`foo::myfunc`
(`anotherClass` near&) →
`@foo@myfunc$qr12anotherClass`

A character 'x' or 'w' may appear after 'p', 'r', 'm' or 'n' to denote a constant or volatile type qualifier, respectively. The character 'q' appearing after one of these characters denotes a function (whose arguments follow in the encoded name, up to the appearance of a '\$' character, which is in turn followed by a return type). So,

`foo::myfunc`
(`const` char near*) →
`@foo@myfunc$qpzxz`

`func1(const int) →`
`@func1$qxi`

`foo::myfunc`
(`int` (`near*`) (`int, int`)) →
`@foo@myfunc$qpqii$ii`

Array types are encoded as 'a', followed by a dimension encoded as an ASCII decimal number and a '\$', finally followed by the element type:

`foo(int (*x)[20]) →`
`@foo$qpa20$i`

Encoded arguments are concatenated together in the order of appearance in the function call. Where a number of identical non-built-in types are function arguments, the character 't' followed by an ASCII character is used for encoding the arguments. The ASCII character denotes which argument type to duplicate, and ranges from '1' to '9' and 'a' onward.

`post::func1`
(`double, double, double,`
`int, int, int,`
`long, long, long,`
`char near*, char near*,`
`char near*) →`
`@plot@func1$qdddiilllpzctata`

The duplicate 'ta' character sequences at the end of the encoded name denote the tenth argument, encoded as 'pzc'.

Why Turbo C++ is better

The encoding scheme used by Turbo C++ represents an improvement over the scheme used in AT&T's *cfront* compiler. The *cfront* scheme was outlined in a paper by Bjarne Stroustrup titled *Type-safe Linkage for C++*, and was largely incorporated

Encoding of arguments

The number and variety of combinations of function arguments make argument encoding by far the most complex part of name mangling.

Argument lists for functions begin with the character sequence '\$q'. Type qualifiers are then encoded as shown in the following table given in Figure 2. Encoding of built-in types follows that for applicable type qualifiers, in accordance with Figure 3. Finally, encoding of non-built-in types follows that for applicable type qualifiers (See Figure 4).

The appearance of one or more digits indicates that an enumeration or class name follows; the value of the digit(s) denotes the length of the name. For example:

into *The Annotated C++ Reference Manual* by Ellis and Stroustrup. Far from claiming its scheme to be a 'standard', the latter work notes that it would be unwise to use identical encodings of type signatures in systems that are not link compatible, which currently is the case among C++ compilers for DOS. There are several ways in which the Turbo C++ encoding scheme improves on the cfront scheme.

First, the cfront encoding scheme was designed with C as an intermediate language. Using this scheme, it is easy to look at an intermediate C file and determine whether a name is encoded, and to decipher any additional information. This feature is of limited use, however, when using a native-code compiler such as Turbo C++ since there is no intermediate C code and the need to view mangled names occurs only rarely.

Second, the cfront scheme is intended for systems that have a single type of function call and a single pointer type. C++ compilers for DOS must deal with the keywords *near* and *far* used in function and pointer declarations, and so must depart from the cfront scheme to be workable.

Third, using the cfront encoding, it is possible (though improbable) to create names in C that, when encoded, look like C++ mangled names. For example, the C variable declaration

Using the cfront encoding, it is possible to create C names that look like C++ mangled names

```
int foo_Fi;
```

would be encoded as 'foo Fi', which corresponds to the C++ encoding for the declaration

```
int foo(int);
```

which will offer problems later at link time. This problem will not occur using the Turbo C++ encoding scheme.

Finally, a scheme like the one used in Turbo C++ makes it easier (and faster) for associated tools such as debuggers to find and process mangled C++ names. This ability to support tools is of particular importance for the future of C++ and object-oriented programming.

Conclusion

In this article, we've taken a detailed look at the Turbo C++ name mangling scheme. The mangling scheme used by Turbo C++ takes advantage of the native-code nature of the compiler and the MS-DOS environment in which it operates, and expedites the use of tools in the development of C++ programs.

EXE

Alex Lane is Senior Technical Writer for the Languages Business Unit at Borland International in California. He is also a regular contributor to the monthly newsletter *The C++ Report*.

Microprocessor Development Software Tools

- Cross Compiler
- Cross Assembler/Linker
- High Level Debugger
- Simulator
- Multi-Tasking Executives
- Floating Point Libraries

Available for the following processors:

Motorola 68000/20/332/302, 68HC11,
6809, 6805, 6801
Hitachi H16, H8/300, H8/500, 6301
Intel 80x86, 8051

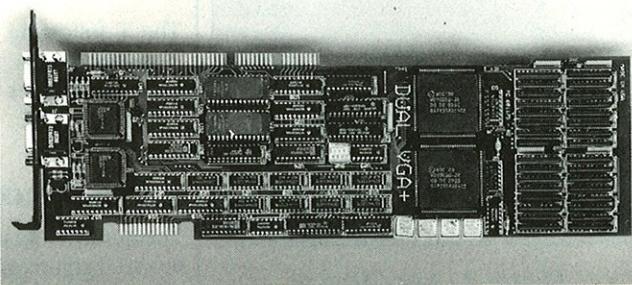
Hosted on MS-Dos,
VAX/VMS, VAX/ULTRIX, SUN, HP9000

Creative Data Systems Ltd.

P.O. Box 180, Basingstoke, Hampshire RG21 1GA
(Tel) (0256) 840646. Fax: (0256) 810131

CIRCLE NO. 514

MULTIPLE SCREEN CAPABILITIES USING A SINGLE PC . . . THE DUAL VGA PLUS



- ★ UP TO 800×600 RESOLUTION
- ★ TWO INDEPENDENT SIMULTANEOUS DISPLAYS WITH OVERLAY CAPABILITY.
- ★ 100% COMPATIBLE WITH IBM VGA/EGA/CGA/MDA VIDEO STANDARDS
- ★ 4 BOARDS IN ONE SYSTEM GIVE 8 SIMULTANEOUS DISPLAYS

NOW INCLUDES WINDOWS 3.0 DRIVER AND MULTIPLE TERMINAL EMULATION PACKAGE

CEBRA COMMUNICATIONS LIMITED

26 LORNE PARK ROAD, BOURNEMOUTH
TEL 0202 299048 FAX 0202 299192

CIRCLE NO. 469

BITON
THE
CLIPPER
ORACLE
LIBRARY



CLIPPER - BITON - ORACLE GIVES ACCESS FROM PC TO
ORACLE ON - MAINFRAME - MINI - OS/2 OR STANDALONE PC.
BITON 1 for summer '87 - BITON 55 FOR CLIPPER 5.

INTERESTED IN BITON?

RING: UK (0727) 50658 FAX: UK (0727) 830111

CLIPPER IS A TRADEMARK OF THE NANTUCKET CORPORATION. ORACLE IS A TRADEMARK OF THE ORACLE CORPORATION.



Inside the Indexes

Comparisons of dBASE products tend to concentrate on such points as the programming language commands and data presentation.

Ken Sawyer gets down to the heart of the matter: the design of the index files.

Most database products use a B+ Tree for maintaining database information in sorted orders. dBASE, Clipper and FoxPro are no exceptions. However, interesting choices can be made on the implementation details.

The most critical choice is the block size. This is the number of bytes read from disk or written to disk when manipulating the B+ Tree. The whole point of using a B+ Tree (rather than a much simpler binary tree) is that it minimises the number of times the disk is accessed. This is important because, even with the latest drives, accessing disk storage takes thousands of times longer than accessing RAM memory. As the block size becomes larger, it is necessary to access the disk less. However, there is a trade-off because large block sizes can require more RAM memory and it can take longer for any given disk access.

Regardless, the most interesting point is that dBASE III NDX index files are slower than Clipper NTX files and dBASE IV MDX index files. This is because the block size of dBASE III NDX files is fixed at 512 bytes, the block size of dBASE IV MDX files defaults to 1024 bytes, and the block size of Clipper NTX files is fixed at 1024 bytes.

The choice of 512 bytes for a block size under MS-DOS is particularly bad. This is because MS-DOS always writes 1024 bytes at a time on 1024 byte boundaries. Consequently, if you tell MS-DOS to write bytes 0 through 511 to disk, MS-DOS will read bytes 0 through 1023 from disk, modify bytes 0 through 511 and then write bytes 0 through 1023 to disk. This means that it would have been much faster to tell MS-DOS to write bytes 0 through 1023 than to tell MS-DOS to write bytes 0 through 512. Similarly, if you tell MS-DOS to read bytes 0 through 511 from disk, MS-DOS will read

bytes 0 through 1023 and then give you bytes 0 through 511.

Clipper's design adds unnecessary complexity, forces a greater code size, wastes disk space and valuable RAM

Note that under OS/2 Version 1.0 the magic number changes from 1024 to 2048. This means that under OS/2 Version 1.0, 2048 would be a better block size than 1024.

There are some other significant differences in the internal index file formats. Within an index file block, Clipper NTX files implement another level of indirection. Rather than just having a list of sorted keys, there is a list of random ordered keys followed by a list of integer pointers to the keys. These integer pointers specify the key ordering. The idea is that it is faster to move and manipulate the integer pointers within memory than the simpler approach of moving the keys themselves.

My intuition says that this was a poor design decision. This design adds unnecessary complexity, forces a greater code size, wastes disk space and wastes valuable RAM. Spare memory can be used to increase performance in many different

ways. One of the best ways is to buffer disk information to reduce disk I/O.

A date with Clipper

Another thing that Clipper does is represent keys entirely as ASCII. For example, under dBASE, a date key is saved as a floating point double representing a Julian day. Under Clipper, a date key is held as eight characters. The date July 4th, 1990 would be '19900704' - the same way a date is stored in the dBASE DBF data file. The Clipper method is superior because no translations from the data files are necessary, and because it is quicker to compare eight ASCII characters than a floating point double.

Finally, many people assume that the FoxPro IDX file format is superior to dBASE and Clipper because FoxPro has a reputation for speed. However, FoxPro has good performance because of its internal memory management. It is optimised for single user performance. In a single user environment, it is possible to work entirely in RAM memory as long as there is enough memory available. Unfortunately, this is not possible in a multi-user environment on a network. In a multi-user environment, it is necessary to keep reading information from disk because it gets changed by other users.

For best performance when using a B+ Tree, the implementation details are critical. These details include disk file formats as well as internal memory management.

EXE

Ken Sawyer has participated in the development of Code Base 4.2, Sequiter's C library which implements complete dBASE and Clipper functionality. You can contact Sequiter Software Inc on 0101 403 448 0313.

When users aren't sure what they want, you could do with a crystal ball.

Let's face it, we all have to deal with users having only a hazy idea of what they want from an Expert System. Not surprisingly, it leaves you and your team uncertain about exactly what needs to be produced.

Fortunately XpertRule will help you to read their minds.

Whether you want to build a complex Expert System or simply add intelligent rule-based modules to a current application, XpertRule is an indispensable part of your toolkit. It includes powerful knowledge structures and rule induction facilities, enabling rule-based knowledge to be represented, captured and tested with ease. Once captured, the rules

are generated in a choice of high level languages.

And knowledge-based applications developed with XpertRule can be ported across different hardware platforms — PC, mini or mainframe — since the generated "source code can be compiled for a range of operating systems.

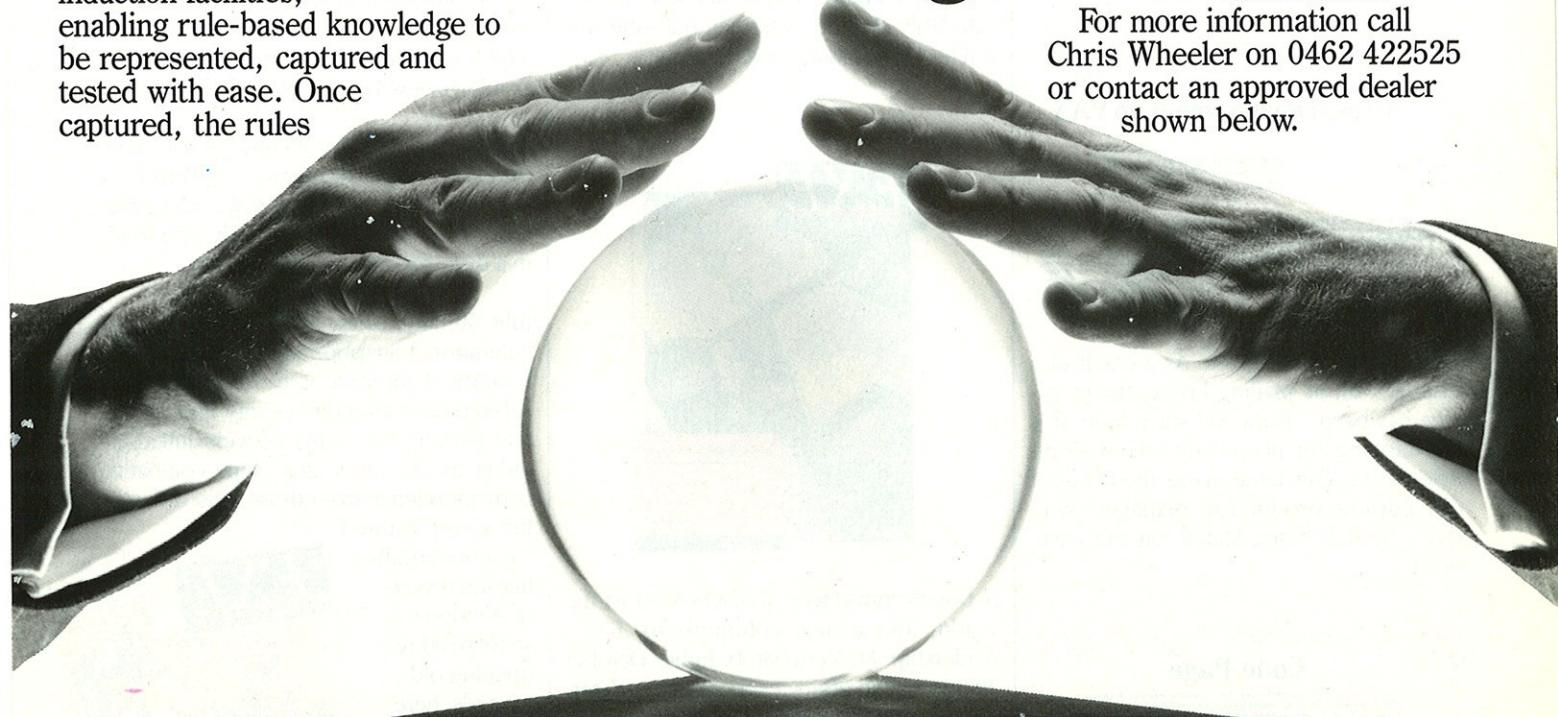
Unlike other Expert Systems, there's no need to learn a new procedural language either.

XpertRule runs on any IBM PC/XT/AT or compatible, under PC-DOS or MS-DOS 2.1 or higher with a minimum of 256K Ram (640K recommended) and a hard disk drive. It can generate code for Pascal, C, Cobol and RPG/400, and also has interfaces to Crystal and the Aion Development System (ADS) expert system shells.

In the future you'll see the XpertRule has helped you to build better applications much faster. Look into it.

For more information call Chris Wheeler on 0462 422525 or contact an approved dealer shown below.

XpertRule worth looking into.



APPROVED DEALERS

Absolute Computing Limited, Absolute House
124 Leavesden Road, Watford, Herts. WD2 5EG. Tel: (0923) 242570

Computer Solutions Limited, Canada Road, Byfleet
Surrey KT14 7HQ. Tel: (0932) 52744

Grey Matter Limited, Prigg Meadow, Ashburton
Devon TQ13 7DF. Tel: (0364) 53499

Roundhill Computer Systems Limited, Orchard House
Ogbourne St. George, Marlborough, Wiltshire SN8 1SU. Tel: (0672) 84535

Reflex Technology Limited, 9 Buckingham Place
Bellfield Road, High Wycombe, Bucks HP13 5HW. Tel: (0494) 465907

Frontline Distribution Limited, Intec 1, Wade Road
Basingstoke, Hampshire RG24 0NE. Tel: (0256) 463344

System Science, 3-5 Cynthia Street
London N1 9JR. Tel: 071-833 1022

Software Express, Portmill House, Portmill Lane
Hitchin, Herts SG5 1DJ. Tel: (0462) 422525

Software Generation • Portmill House, Portmill Lane, Hitchin, Herts SG5 1DJ

• DEALER ENQUIRIES WELCOME •

CIRCLE NO. 471

Now we are Five

This Magazine is five years old this month. We look back at some of the highlights (or otherwise) of its existence.

The first issue of .EXE is published in February 1986. The cover feature is an introductory article to Intel's new 80386. There are also pieces on developing for DR's GEM, a survey of compilers costing less than £100, and a general feature on windowing environments which compares GEM, Microsoft's Windows and Epson's TAXI. (*Wot that? Ed*) Ian Adams' editorial promises '... we provide a forum for you to exchange views and problems.' The magazine contains no program code.

Cross Development on VAX

Roger Farrington believes embedded systems programmers should be using cross-assemblers and compilers on host systems.

There are a great many choices to be made for an embedded system. The choice of development medium, operating system, hardware, memory, disk, and so on. Many developers start with assembly language, because it is the easiest to learn. They then build up from there, and may add a 'C' or 'Pascal' interface later.

High-level languages are rising in popularity, and more standard operating systems are available for them. However, they are highly machine dependent, as they are usually written to make the most of the hardware.

Yet even with these, there will usually be a 'port' to do.

It is a complex task, but it can be done.

It is a complex task, but it can be done.

Issue 5 is in October '86. As a result of a dispute with its mailing house, the magazine has been 'off the air' since June. It is still thinking big, however, as this VAX article shows. This issue marks the debut of our current production manager, who styles herself 'Katey Matey' on the masthead.

Code Page

The Code Page is an occasional series for low-level software hackers. We want to provide snippets of code or information which give a greater insight into the workings of a particular machine or piece of software. This month, Robert Schifreen delves into the innards of MS-DOS's PSP.

When DOS is asked to load a program, it does so by locating the block of memory to hold the program. Memory is then mapped to build a header in the first 48KB bytes of memory. Then, the program itself is loaded, normally starting directly after the header at address 1000h.

The header is known as a Program Segment Prefix, or PSP. This data structure is used by DOS before and during the execution of a program, and is therefore well documented in books like Norton's

Programmer's Reference Guide. It is also described in the DOS 3.0 manual.

For example, if you wanted to convert DOS 3.0 to DOS 4.0, you would need to change the PSP to reflect the changes in memory layout.

Enough of the more useful parts of the PSP were undocumented and obscure, so I decided to write a program to dump them onto a screen or printer. For safety, I made it a simple drop-in boot drive.

Everything is known to be correct for DOS 3.0.

If you would like to experiment, the easiest way is to convert DOS 3.0 to DOS 4.0 by language. That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

For safety, I made it a simple drop-in boot drive.

We welcome technical tips from readers.

Please send them to us, and we'll send you the EXE Contributor's Guide.

That program can access

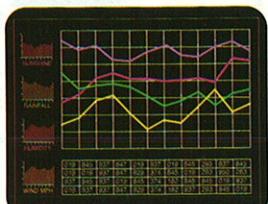
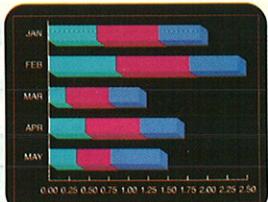
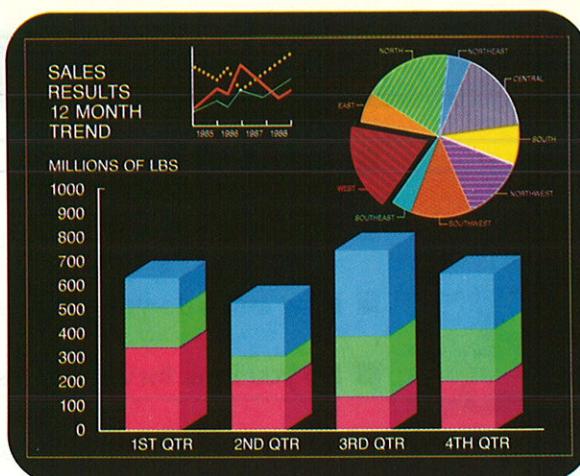
the PSP.

The PSP is simply reading bytes in the region of CS to 0FFFFh. It can then print them out, or save them to a file.

dGE VERSION 4 GRAPHICS FOR dBASE

**BITS PER
SECOND**

BITS PER SECOND LIMITED
14 REGENT HILL BRIGHTON BN1 3ED
TELEPHONE: 0273 727119
FAX: 0273 731925



dGE extends the database language of your choice to give full colour graphs and charts from **within** your programs.

dGE version 3 was voted the "best graphics tool for the database manager" for the second year running by readers of the influential U.S. magazine DataBased Advisor.

Now Version 4 is here with even more features:

- SuperText with variable sizes and face styles
- SuperIcons loaded from file
- SuperText and SuperIcon editor that can import PCX images
- Built-in Vector Text functions
- Gantt charts
- XOR capability added to many functions
- New interfaces to Clipper 5, FORCE and Vulcan
- Mouse "hot" regions
- A Norton Guide to functions

dGX, the Graphics Design Centre, has been completely revised, adding:

- Sophisticated database query and summary
- Dynamic scaling
- Import of PCX images
- Statistics functions
- Gantt charts, hi-lo and 3-D bars

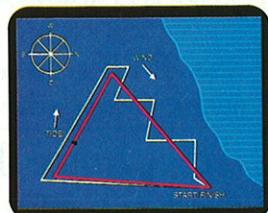
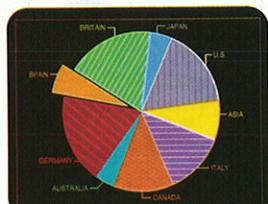
Version 4 comes complete with all the interfaces to dBASE and its dialects, the 'C' language, R:BASE compiler and Vulcan.

dGE supports CGA, EGA, VGA and Hercules modes and will output images to matrix printers, lasers and plotter.

Call for information.

Price £245 + VAT direct from Bits Per Second Ltd or from your authorised dealer.

dGE is a TRADEMARK of BITS PER SECOND LTD. OTHER BRAND AND PRODUCT NAMES ARE TRADEMARKS OF THEIR RESPECTIVE HOLDERS.



CIRCLE NO. 511

**Bring Graphs
and Charts to your
Windows Applications
Licence-Free with
Graphics Server™**

Introducing the **Graphics Server Software Development Kit (SDK)** from Bits Per Second. The most cost-effective, time-efficient way to integrate full-colour graphs and charts into your Windows applications. Here's why.

Client-Server Model

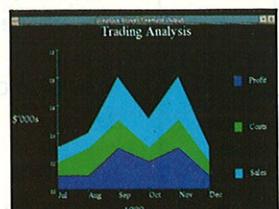
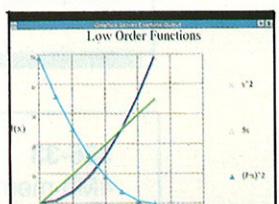
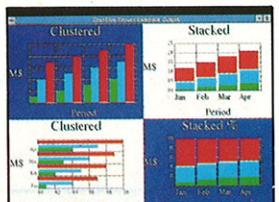
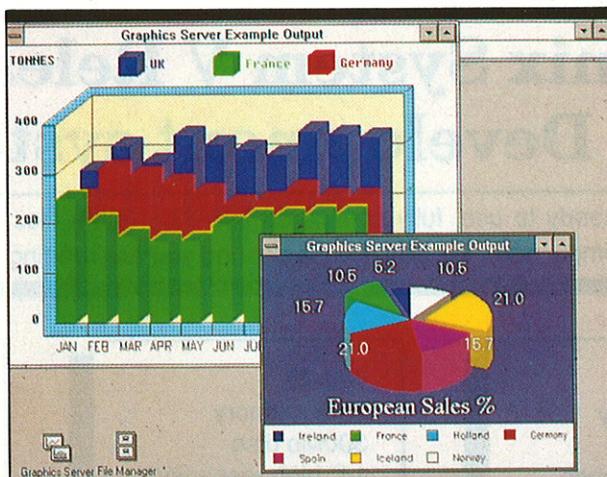
Graphics Server is a Windows server process which responds to multiple, concurrent requests for interactive graphing and charting functions from client applications. It enables graphs and charts to be incorporated into any Windows application having a Dynamic Link Library (DLL) or Dynamic Data Exchange (DDE) capability. This includes C programs, Superbase, SQL, Windows, ToolBook, Word for Windows and many others. In fact, virtually all Windows applications!

Multiple Choice

Graphics Server lets you choose from 16 graph types, including 2-D and 3-D bar (simple, stacked and clustered), 2-D and 3-D pie, area, Gantt, scatter, polar, log-log/log-linear, line, high-low-close, X-Y and polar, plus a variety of statistical functions.

Ease of Programming

Graphics Server, through its multi-level interface, provides the level of control you want. In as few as eight calls you can open a new graph window, define titles, legends and labels, transfer data, draw a chart, and output the chart to the clipboard or printer. Graphics Server takes care of all the details; scaling the graph to match the data, matching fonts, text size and position to the graph, even resizing and repainting the graph window when necessary! You'll save development time. And money.



Alternatively, if you want full control yourself, Graphics Server provides over a hundred functions to help you design, create and manage virtually any kind of graphically drawn image. Multiple graphs, multiple windows, user interaction via hot regions. All these and more!

Full Support

The Graphics Server SDK gives you all the tools, documentation and examples you'll need to make immediate and effective use of Graphics Server. Within hours your applications will be transformed by the inclusion of presentation-quality graphics.

Telephone/fax support and a bulletin board service are provided for all registered users, and for power users requiring an even higher level of service and security there's a special VIP Support Programme.

No Licences!

The Graphics Server SDK costs £275 + VAT. No further licence fees to pay.

To place an order, request a demo disk, or obtain additional information, call . . .

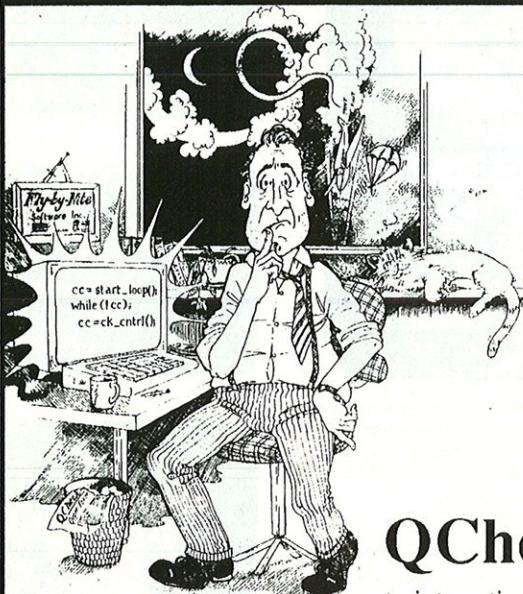
All product names are trademarks of their respective holders.

**BITS PER
SECOND**

Bits Per Second Ltd
14 Regent Hill
Brighton BN1 3ED
Tel (0273) 727119
Fax (0273) 731925

CIRCLE NO. 472

THE SOFTWARE TESTING NIGHTMARE ?



- Are you serious about the way you develop programs?
(Of course I am)
- Do you test your programs?
(Doesn't everybody?)
- How much do you test your programs?
(I try out all, well most, of its functions)
- Do you look forward to testing your programs?
(Who does?)
- Can you quantify how much you test your programs?
(How could I, even if I wanted to?)
- Have you considered the implications of not being able to demonstrate that you have fully tested your program?
(...Ah, um...not really yet)

QCheck is a unique program that enables professional software developers

to interactively find out how much their program has been exercised. It also provides objective evidence in the form of listings showing unexecuted lines. For DOS high level language compilers. A single user licence is £150.00+VAT (*how much does just one undetected bug cost you?*)

Also - ETP an Epson™ to Postscript™ translation utility that allows existing applications to print to your Postscript laser printer, whether connected locally or on your network. A single user licence costs £55.00 + VAT.

Telephone 0883-341697, Fax 0883-341343

Seltek Ltd., Seltek House, 38 Westway, CATERHAM, Surrey. CR3 5TP.

SELTEK

CIRCLE NO. 515

Unix System V Release 4.0 Development systems

Example:- Ready to use, fully configured 386-33, 100Mb disc and Unix V.4 £2995.00...
simply open the box, switch on, and you are running Unix.

386-33
4Mb memory
100Mb disc
14" paper-white
monitor
Desktop cabinet
Unix V.4

£2995.00

486-25
8Mb memory
300Mb disc
14" high resolution
colour monitor
16bit ethernet card
Tower cabinet
Unix V.4

£6995.00

All systems include as standard:

- Compatibility with Xenix, BSD and V.3.x
- Full conformance with the POSIX standard of the IEEE
- ANSI standard C Compiler and full developers toolkit
- X Windows V11.3 with UI/Open Look and OSF/Motif GUIs
- NFS and RFS file systems
- PC Interface server for integrating PCs over a LAN or RS232 link - a real alternative to Novell.

All our systems include a full implementation of Unix System V Release 4.0, keyboard, mouse, serial and parallel ports, documentation and 1 year warranty.
(other systems and options available)

NB These special offers are a 5% discount on our normal list price. Developers and VAR enquiries for these and all 7x Unix and Telecommunications Systems and Services are welcome.

**Call 7x - the Unix experts on
0344 872684**

7x Computer Systems,

TTC, Silwood Park,

Ascot, Berks SL5 7PW

Tel: 0344 872684

Fax: 0344 872276

Tlx: 849645 SEVENE G

CIRCLE NO. 473

A new Smalltalk

Steve Cook has a look at Objectworks\Smalltalk Release 4, the new Smalltalk from ParcPlace Systems, now integrated with native window managers.

Smalltalk originally dates from research done at the Xerox Palo Alto Research Center in the early 1970s. At PARC, Smalltalk was the programming environment for Alan Kay's vision of personal computing - the 'Dynabook', a small networked handheld personal computer with a high-resolution display. Smalltalk first became widely known in August 1981, when BYTE magazine featured a collection of articles about the system. Over the next couple of years, commercial implementations of the version called Smalltalk-80 became available from Tektronix, Apple and ParcPlace Systems. Tektronix and Apple subsequently dropped out of the Smalltalk business, while ParcPlace Systems (which spun out of Xerox specifically to develop Smalltalk) has continued to produce versions of Smalltalk-80, the most recent being version 2.5.

Smalltalk is not easy to learn. Apart from its sheer size, it demands a shift of perspective which can be difficult to grasp for people used to more conventional programming technology. It is quite usual to spend several months getting familiar with the system; becoming a real expert can take years. Up to now, the most effective use of Smalltalk has been in research laboratories, prototyping novel ideas (Randall Smith's Alternate Reality Kit is a vivid example).

Recently, with the appearance of window managers on host platforms, the Smalltalk-80 policy of taking over the entire machine and managing its own windows has become less and less appropriate. The original window manager was written under the assumption that the only window allowed to change is the active one. Windows updated by asynchronous processes, such as clocks or incoming data from a network, would simply splatter updates through any windows overlapping them. This limitation persists in Version 2.5 and, together with the 'flap-out' scrollbars and absence of colour, makes it appear eccentric and obsolete in a modern PC or workstation environment. Digitalk's Smalltalk/V, a less sophisticated system targeted at the PC and Mac marketplace, is better integrated with host window systems, and provides an attractive and cheaper alternative.

the assignment symbol is now the familiar := (in the previous version a left arrow symbol could also be used). Smalltalk programs are structured in terms of 'sending messages to objects'. Each message, when it arrives, causes the execution of a corresponding 'method' defined in the object's class. The dynamic association of methods with messages, often called 'dynamic binding', is the feature which gives Objectworks\Smalltalk much of its power.

Objectworks\Smalltalk Release 4 is not for creating shrink-wrapped applications

Now comes Objectworks\Smalltalk Release 4, the new Release from ParcPlace, available now on Apple Macintosh and on several UNIX systems under X, soon to be released under Microsoft Windows 3.0. Objectworks\Smalltalk Release 4 discards the old Smalltalk-80 window manager and instead uses the platform's native window manager for handling its windows.

I have been programming in Smalltalk-80 for a total of seven years, and have built some fairly large systems in it. My burning question, when I got my hands on Objectworks\Smalltalk Release 4, was whether at long last this is a Smalltalk-80 system for building portable, fully-integrated workstation applications as well as research prototypes. The answer is yes, but with some strong reservations. This preview is based on my experiences with Objectworks on an 8 MB Macintosh IIx.

Smalltalk programming

Most of Release 4 is very familiar. The Smalltalk language is almost unchanged, except

Programming in Objectworks is not like programming in C or Pascal. There is no main program, no apparent compilation and no linking. Instead, the programmer alters an Objectworks image incrementally, changing existing methods and introducing new ones. At any point the image can be saved in a file. It can subsequently be restarted, recreating the same situation as when it was saved, including window placement and (with some exceptions) window contents.

Although Objectworks feels just like an interpreted system, it actually compiles methods into machine code on the fly. This means that an Objectworks image developed on a 386 PC can be run on a 68020 Macintosh without alteration. When running on the PC, the run-time system compiles methods into 386 machine code; on the Mac, 68020 code will be produced. All of this is completely transparent to the programmer.

The original Smalltalk was designed to be used by a single person. Smalltalk-80 provided tools for managing projects and changes, but these are only usable for groups of two or three programmers. For projects of larger size, additional third party tools are necessary. There are a few companies (such as Instantiations Inc. of Portland, Oregon) making such tools for Smalltalk-80, and with luck these will be updated to work with Release 4.

Apart from the different look of the windows, and a few additional options here and there, the programming tools in Release 4 are just as they were in the earlier system. The system browser is used for navigating around the class hierarchy, reading, creating and editing code. The workspace is used for saving and evaluating code fragments. Any object can be inspected, with a window giving access to its state. The debugger was always one of the best around, and remains so. The file list, file editor, system transcript and change list browser complete the repertoire of tools.

Earlier versions of Smalltalk used a pop-up menu on the screen background to launch new windows and utilities. In the native windowing environment, Smalltalk has no access to the screen background, and this menu is replaced by a window called the *launcher*. The trouble with the launcher as supplied is that it is easy to lose under mounds of windows. To fix this, I spent a few minutes adding a menu item to each open window, allowing me to 'unbury' it. There are not many systems around which are open enough to allow changes to be made so easily at such a fundamental level, and this strength of Smalltalk is preserved in Release 4.

Digging deeper

Digging into the code revealed substantial differences between Release 4 and the earlier systems. Some familiar classes, like Form and BitBlit, have vanished completely, to be replaced by Image, Pixelmap and RasterOp. The Form Editor and Bit Editor have gone too. There is no longer a global called Display; in Release 4 everything must be done via a host window. There are many new classes supporting colour, and the user interface classes are very different.

On the other hand, many parts of the system remain the same. Numbers, Collections and most of the Kernel and System class categories are unchanged. Exceptions and Processes are just as they were. It took me just a few hours to understand how Release 4 differed from its predecessors, and although the differences are substantial, the overall feel of the system is extremely similar.

Smalltalk's user interface is structured using a principle called Model-View-Controller (MVC). The basic idea is that the Model is the data to be interacted with, the View is the rendering of that data on the screen, and the Controller provides the mechanisms for mapping input events into changes to the Model. Release 4 retains the MVC principle,

but the view hierarchies and protocols are quite different. Reflecting the fact that some objects exist in the host window server rather than in the Objectworks image itself, Release 4 has Windows as well as Views. Each window on the screen is represented by an instance of the class ScheduledWindow. The parts of windows are represented by classes such as

The debugger was always one of the best around, and remains so

VisualComponent and CompositePart. Each part of a window is held in a Wrapper object, which adds borders and controls, as well as doing translation and clipping. For some reason, the hierarchy of visual classes has been designed so that only non-composite views can scroll. This seems an unnecessary restriction.

In the window management scheme of Release 4, the host window manager does not guarantee to retain the graphical information written onto its windows. It is up to the programmer to provide a method which can fill in the parts of windows exposed by window operations. This is often called 'damage repair'.

Colour

Objectworks\Smalltalk Release 4 introduces full colour graphics. The whole graphics subsystem is completely different from earlier versions, and is dignified with a new name: the Smalltalk Portable Imaging Model (SPIM). Displaying graphics in SPIM is accomplished through graphics contexts associated with graphics media. A graphics context specifies various parameters, including default paint colour, font, and line display attributes. Graphical objects are displayed either by sending messages direct to the graphics context, or by sending messages to graphical objects with the graphics context as a parameter. The result is displayed using a stencil and paint principle, similar to that used in Postscript. Texts, polylines, polygons, arcs, wedges and images are supported.

All co-ordinate systems in Release 4 are pixel-based with no facilities for scaling. This is a step back from earlier Releases which supported both scaled and translated co-ordinate systems.

Some effort has been taken to ensure that colours in Release 4 are portable between platforms. Colours can be specified by red-green-blue (RGB) or by hue-saturation-brightness (HSB) co-ordinates. Pixel images up to 24-bits deep are stored in a portable format, and will appear with the right colours on any machine that Objectworks\Smalltalk runs on. If the colours specified cannot be rendered on the machine, Release 4 will do the best it can, including dithering (a technique which gives adjacent pixels different colours, relying on the eye to average between them).

The previous Smalltalk font facilities were underwhelming. Font technology has developed at an extraordinary rate over the last few years. Release 4 provides a nifty mechanism whereby you can issue a font query to the underlying system, and it will do its best to match what you asked for. If you want a specific font, you can say so and the system will give you it if it can; otherwise you can specify various attributes (like whether or not you want a serif font, and how big and bold you want it) and it will find you something near.

Not included with the basic system, but as a set of 'fileIns' - source code files which you can easily compile into the image using the file list tool - is support for Postscript printing. I did not have a Postscript printer available for this review, so I couldn't try these features out, but from the code they looked fairly comprehensive.

The user interface

Running Release 4 and Release 2.5 side-by-side really shows up how much of an improvement there is. Release 4 looks thoroughly modern in comparison. By judiciously configuring a set of defaults in a class called LookPreferences, you can get nice coloured backgrounds and foregrounds, with a pleasing 3D effect. Conspicuous by its absence is a LookPreferences browser tool, which is now high on my list of 'goodies' to write or find. (Goodies is a term used in the Smalltalk community to describe small public-domain accessories for filling in. These can often be found on various networks.)

The scrollbars are more aesthetically pleasing than the old flap-out ones. They come narrow and a little fiddly, but the width can be adjusted with LookPreferences. A range of buttons is provided, including radio buttons, checkboxes, press buttons etc. These do not have a 3D press effect. I was disappointed to find that the attributes of buttons (text font, size, colour etc) are kept in a system global; it would make more

U.K. VERSION
NOW SUPPORTS BOTH
DOS & OS/2, DUAL MEDIA

POWER AND FLEXIBILITY



BRIEF 3.0 increases your productivity for just £199.00*

BRIEF has established itself as the leading programme editor on the market today. Now, with the launch of the latest enhanced version - BRIEF 3.0 - it offers even better productivity.

ENORMOUS UNDO POWER

BRIEF 3.0's enhanced UNDO capability takes the fear out of programming. BRIEF 3.0 allows you to undo any command that affects the cursor or the text (including global changes) up to 300 times, even back to the beginning of your session. It's the safety net you need to program with confidence.

REAL FLEXIBILITY

There's built-in support for 36 languages including Assembler, BASIC, C, COBOL, dBase, FORTRAN and Pascal. And thanks to its complete macro language you can make BRIEF 3.0 fit your own programming style with ease: change key assignment, commands, prompts, colours - even the indenting styles.

BRIEF 3.0 The Programmer's Editor

Send to: Solution Systems UK, The Maltings, Green Drift, Royston, Herts SG8 5DB

Please rush me my copy of BRIEF 3.0

I enclose a cheque to the value of £228.85 (£199 + V.A.T.)

NAME _____

ADDRESS _____

POSTCODE _____

Please charge my Access/Visa* account by the sum of £228.85 (£199 + V.A.T.)

Card No: _____ Expiry Date: _____ Signature: _____

* Delete as appropriate

No other editor puts this much power and flexibility in your hands. So if you're looking for extra productivity purchase your copy of BRIEF 3.0 today.

**Solution
Systems**

SOLUTION SYSTEMS are the UK distributors for BRIEF 3.0. When you buy from us you get the authorised UK version of BRIEF 3.0 dual media (3½" and 5¼") plus bonus macro diskette.

Special phone line for Access/Visa card holders.
Save time by phoning
0763 244141
between 9.00 and 6.00
Monday to Friday
to place your
order.

N.B Official purchase orders
from local authorities,
PLCs and educational establishments
are accepted by post or by Fax.
Fax no: 0763 244025

Orders despatched same day.

sense for them to be associated with each individual button.

Each pane in a window has a menu bar along the top, and buttoning within this bar brings up the operate menu (corresponding to the middle button with a three-button mouse). This is useful on the Macintosh, because of the one-button mouse; on a system with a three-button mouse, these menu bars are irrelevant. On the Macintosh I would like another area to button in, bringing up the window menu - or better, have the window menu on the normal Mac menu bar. Having to press the command key and click to get at the window menu is not ergonomic.

I was hoping the system was going to be able to adopt the look and feel of the platform it is running on. Ideally, I want to develop an application on the Mac, using Mac scrollbars, dialogs etc, and then just lift it and drop it onto Motif or Windows 3.0 and have it automatically adopt the local user interface components. Release 4 does not do this. It simply uses the host window system to provide it with rectangles to draw in; within these rectangles it looks the same regardless of which system it runs on. This means that an application created under Objectworks will not look like a normal Mac or Windows application. On the Mac in particular, there are only two menus in the menu bar. One is a file menu, which only allows the user to quit the application and, for some reason, to bury the current window. The other is an edit menu, but this was greyed out and I could not find any way of making it active. Cut and paste does not work between Objectworks windows and Mac windows. The scrollbars do not look like Mac scrollbars. Particularly annoying in the Mac context is the absence of the normal resize box in the bottom right-hand corner of each window: instead, you have to command-click in a window and select resize from the resulting pop-up menu.

I found a fileIn which contained some code for creating Mac menus and dialogs, and I managed to make it work, so it is certainly possible to use these underlying facilities; but the code is not portable across platforms. A further fileIn allows PICT files to be imported from the clipboard, and I used this successfully to import an image from MacPaint and display it within Release 4. There is also a fileIn for accessing the Mac's sound resources.

A normal Macintosh application uses entities called resources to implement its windows, menus and dialogs, and it is a pity that Objectworks\Smalltalk provides no interface to these. A programmer wishing to

use them would have to start from scratch, by linking his own primitives into the Objectworks run-time system.

Smalltalk is not easy to learn - it demands a shift in perspective which can be hard to grasp

There is a trade-off between portability and the ability to support the detailed features of each platform. Objectworks\Smalltalk has focused on portability, with great success - you really can take an image developed on one platform and run it on another. It would be nice if there were additional facilities to access the underlying features of each specific host platform, and it would be even nicer if these were organised so that an application would adopt the look and feel of the platform it is on. Still, there is nothing in Objectworks\Smalltalk that will stop you doing these things for yourself if you are brave and have the time.

Memory

Smalltalk has always been strong on memory management. The programmer doesn't need to worry about memory, as it is all managed automatically. There is no way a pointer in Smalltalk can refer to anything except a valid object, and when there are no more pointers to an object its space is reclaimed. Release 2.5 was pretty sophisticated at managing its memory, and Release 4 is even more so. Memory is divided into different zones which are handled according to what kinds of objects live there. Short-lived objects are collected using a technique called scavenging, whereas longer-lived objects are collected incrementally. In Release 4, the incremental garbage collector runs as a background process within the Smalltalk image itself, and various parameters of the memory management policy can be altered by the programmer.

An interesting new feature in Release 4 is the class `WeakArray`, implementing 'weak pointers'. These are pointers which do not inhibit garbage collection. They can be set up to detect when objects die, and run finalisation code. They can be particu-

larly useful for distributed algorithms, running in several communicating images. For example, a distributed garbage collector might want to inform other images whenever an object dies, to update global reference counts.

During several hours of normal program development with Release 4, I didn't notice the system garbage collecting or compacting at all. I did manage to make it work hard by writing some code that repeatedly allocated large pixmaps; this caused the system to stop for a second or so at intervals to garbage collect and compact memory. The system does use a lot of memory. My experience with the earlier version suggests that 8 MB is a practical minimum for writing systems of any significant size.

Performance

On the Mac IIx, my (beta) copy of Objectworks was reasonably responsive, although there was a perceptible delay between pressing the mouse and getting the result. In comparison with some other Mac applications, such as MacWrite, Objectworks was noticeably sluggish, although to be fair I didn't have any full colour applications with which to compare it. Sometimes, for no apparent reason, the system didn't respond and I needed to try twice to get a selection or menu. There were no crashes while I was testing it.

Documentation

The 300-page manual is comprehensive, and covers most aspects of the system in adequate detail. It includes a section on how to port existing Release 2.5 applications to Release 4, grading the impact of the system changes and the difficulty of conversion for various types of application. Applications which use graphics extensively are greatly impacted and are difficult to port, whereas applications using only text are relatively easy.

Much of the documentation is contained within the code itself. I found some of the comments in Release 4 to be incomprehensible, and several to be just plain wrong. For a system as difficult to learn as Objectworks, it can be discouraging to come across incorrect comments, and this is one area where I think a lot of improvement could come from a relatively small amount of effort.

What's it for?

Objectworks\Smalltalk Release 4 is not for creating shrink-wrapped applications. Without a lot of extra work, developments with

DOS PC-NFS UNIX

*If you can't beat it - join it
because you can program it*

- **Sun PC-NFS Programmers Toolkit** - a C-library interface to TCP/IP and NFS protocols
- Berkeley socket library and RPC library
- Microsoft C-compatible with network services (YP) support
- File and print redirection and file locking support in run-time mode
- Also: toolkits/libraries for TCP/IP over OS/2, VAX/VMS and X

Unipalm Limited 145-147 St Neots Road • Hardwick • Cambridge CB3 7QJ • England
 • Tel 0954 211797 • Fax 0954 211244 • Intl. Tel +44 954 211797 • Intl. Fax +44 954 211244



CIRCLE NO. 475

► Learn how to build graphical user interfaces for UNIX software from the experts



OSF/Motif Programmer's Workshop

This popular 4 day workshop concentrates on using the Motif toolkit to build applications with OSF/Motif look and feel. Includes full course materials and extensive hands-on.



X Programmer's Workshop

Xlib, the base layer for all UNIX GUIs, is covered in detail during this 4 day hands-on workshop. Complimentary to Motif course for those writing their own widgets.



X Strategic Overview

An intensive one-day seminar on X looking at marketing, commercial and technical issues. Ideal for sales and marketing personnel or those overseeing company strategy.



On-site Training

For larger project teams, IXI provides these courses on-site. Courses can be tailored to meet your particular needs. Call Kim Fawkes at IXI Sales for more information.



IXI Limited 62-74 Burleigh Street Cambridge CB1 1OJ England Tel: +44 223 462131 FAX: +44 223 462132
 * OSF/Motif is a trademark of the Open Software Foundation, X is the trademark of MIT.

T

R

A

I

N

I

N

G

CIRCLE NO. 512

Objectworks will have their own individual look and feel, and will not integrate readily into the native environment. The product comes into its own when portability across platforms is more important than integration with the native environment. In this case, Objectworks is probably the most portable system around.

Traditionally, important application areas for Smalltalk have been in rapid prototyping, and in data analysis and modelling in large business applications. Objectworks\Smalltalk Release 4 is a substantial improvement on earlier versions, and makes Smalltalk an even more attractive proposition in these areas where it is already strong.

The next step would be real integration with host systems, combined with binary portability, and these improvements would open up new markets for the product.

Over the years, Smalltalk has been one of the most influential pieces of software in existence. Smalltalk gave the world windows, mice, pop-up menus, mode-free interaction and object-oriented programming. It is curious that such an influential system should now be catching up with its own legacy.

EXE

Steve Cook is Managing Director of Object Designers Ltd, specialists in the introduction and application of object-oriented technologies. He has used Smalltalk-80 since 1983. He can be reached on 0279 755396.

Objectworks\Smalltalk Release 4 can be obtained from AI International (0923 247707), price £2950 for all platforms. It is available now for Apple Macintosh, Sun SPARCstations and Digital DECstations. It will be available in Q1 1991 for IBM RS/6000, HP 9000 Series 300, Apollo workstations and MS-DOS with Windows 3.0.

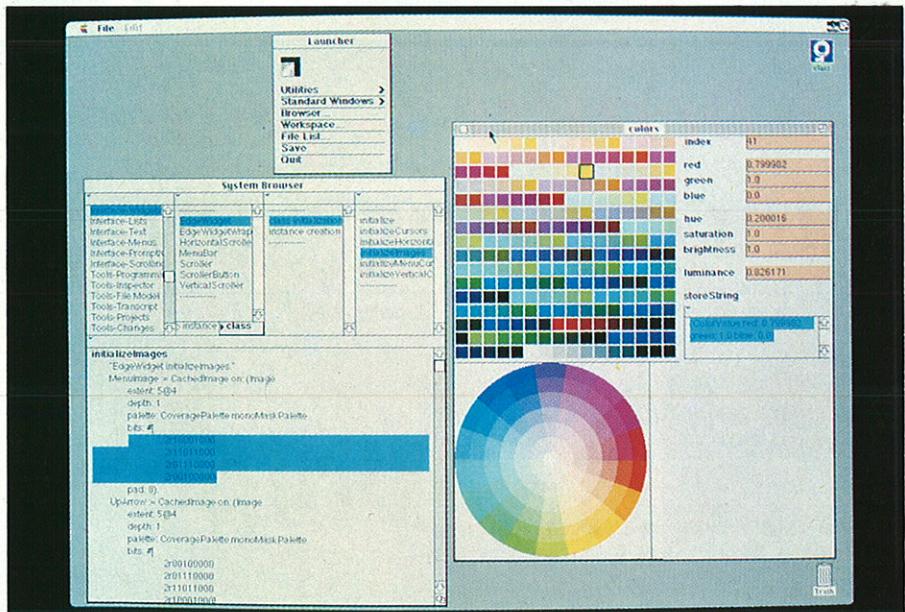


Figure 1 - Objectworks\Smalltalk Release 4 screen display on Mac

FORTRAN

FTN77 -

The Salford FTN77/386 and 486 compilers use DOS extender technology to turn your 386SX, 386 or 486 based PC into a powerhouse. FTN77 programs run under MS-DOS, alongside your existing applications, but what a difference! Execution speed on a 486 is comparable to a top-of-the-range VAX 8650, and compilation is even faster!

- No 640K limit. Accesses a linear **4 Gigabyte** address space.
- Virtual Memory support.
- Supports Intel and/or Weitek numeric coprocessors.
- Emulates 80387 if no coprocessor available.
- Lightning fast Execution. Uses the full 32 bit 80386 instruction set.
- Lightning fast Compile and Link. Nothing else comes close...
- ANSI validated by the NCC and US Dept of Commerce.
- Many extensions (e.g. F66 compatibility, inline assembler)
- Extensive run-time library including many system facilities.

FTN77 is more than just a number-cruncher - it's a **bug-cruncher** too! The integrated source level debugger combines remarkable power with an intuitive window based user interface. Just as importantly, the optional run-time checks on interfaces and the use of uninitialised data detect bugs that very few other compilers or analysis tools can spot.

FTN77 is compatible with numerous graphics packages, including the powerful and portable INTERACTER library from ISS. Please call for details and a **free** demo disk.

"For sheer speed and sleekness, the (Salford) compiler stands out like a Ferrari among Yugos".
(Computer Language Magazine, May 1990).

Polyhedron Software Ltd., Magdalen House,
98 Abingdon Road, Standlake, Witney, OX8 7RN, England
Tel (+44) 0865-300579 Fax (+44) 0865-300232

All trademarks acknowledged



CIRCLE NO. 476

CopyControl

THE NEW GENERATION OF COPY PROTECTION

NO	User hassle
NO	Back up problems
NO	Hardware add ons or special disks needed
NO	Changes to source code required
YES	CopyControl beats ALL bit-copier Programs
YES	Floppy disks, hard disks and networks supported
YES	CopyControl is totally transparent to the user
YES	You can limit program use by no. of copies, no. of executions or date
YES	CopyControl works on all IBM compatibles
YES	Free demo disk available

For Further Information Phone or Write to:

microcosm

Microcosm Limited,
17 Cranbrook Road, Bristol BS6 7BL
Telephone: 0272-441230

CIRCLE NO. 477

MULTIPLE INTELLIGENT COM PORTS FOR OS/2

THE IMPAC 8 & 16 INTELLIGENT COMMS CARDS

- * MICROSOFT COM X COMPATIBLE
- * 8 TO 64 FULL MODEM PORTS
- * REDUCE CPU OVERHEAD FROM 40% TO 1% PER PORT
- * FITS ALL ISA (PC/XT/AT) MACHINES
- * DOES NOT USE SHARED RAM
- * PROVEN FIELD PERFORMANCE
- * UK DESIGN AND MANUFACTURE
- * DRIVERS FOR DOS, OS/2, UNIX & XENIX

DEALER ENQUIRIES WELCOME

For more information contact:
SCL Downswood, Carlton Road
Reigate, Surrey RH2 0JQ
Telephone: 0737-762200

CDOS DRIVER ANNOUNCED

CIRCLE NO. 478

What Prolog programmers have been waiting for!

Prolog-2 for Windows

version 3.0

Once again Expert Systems Ltd. are first to deliver a Prolog system matched to today's programmer's needs.

Our new Prolog-2 for Windows 3 is the only package that allows programmers to combine the power of Prolog logic programming with the acclaimed GUI of Windows 3.

Now you can create 32-bit applications for the world's favourite GUI with your favourite programming language.

Just look what Prolog-2 for Windows 3 gives you:

- Full 32-bit addressing using 386 enhanced mode, no more limits on workspace, heap etc.
- Virtual memory support
- Startling speed. Up to 150 kLIP on a Research Machines VX486
- Full bidirectional C interface for embeddability
- Integrated window editor and graphics
- Over 500k of Hypertext help
- Workstation performance on a PC for just £1495.00

I need a Prolog to write Windows 3 applications right now. Please rush me Prolog-2 for Windows 3.

Please add £10 p&p plus V.A.T.

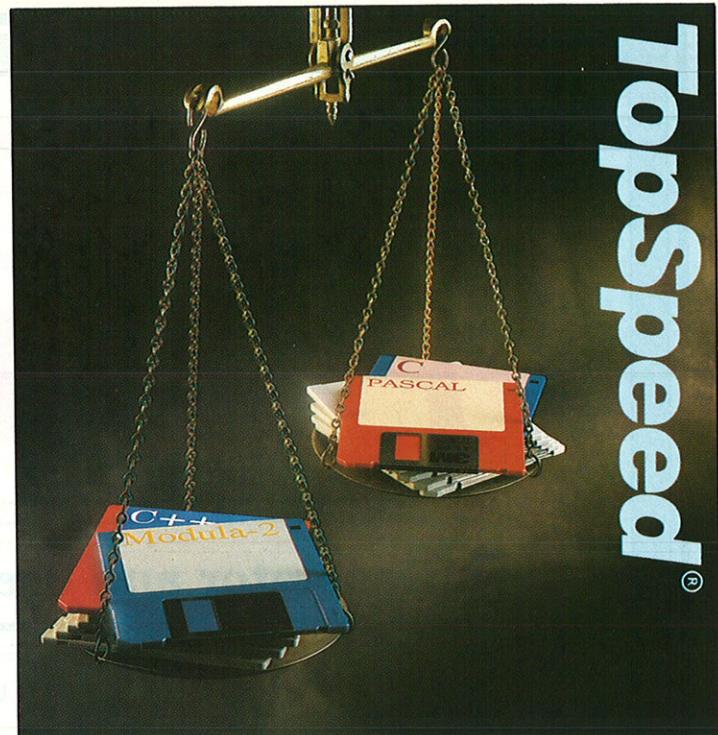
My cheque is enclosed or My official order is attached or I wish to pay by Mastercard/Visa
All trademarks acknowledged

copies at £1495 (£747.50 acads)	
Card no.	
Expiry Date	
Name and address	
Signature	

Expert Systems Ltd.

Unit 12, 7 West Way, Oxford, OX2 0JB. England. Tel 0865 794474 fax 0865 250270

CIRCLE NO. 479



TopSpeed®

Are you weighing up the pros & cons of your next choice of language?

Will the language you choose today meet your development needs of tomorrow? How can you save the time and money already invested in past projects? Can you make the most of the skills you have now, and yet still keep up-to-date with developments in programming technology? A fine balancing act by any standards.

DOS or OS/2, a first time user or a professional developer, the choice has never been easier. TopSpeed, from Jensen & Partners International, is a growing family of languages which share the same code generator, linker and run time system, as well as the much acclaimed TopSpeed development environment. All TopSpeed languages simply slot in to form a *true* multi-language programming environment.

Whatever your choice of languages, now or in the future, keep the balance in your favour! Contact JPI on 071 253 4333 or write to Jensen & Partners UK Ltd, The Turnmill, 63 Clerkenwell Road, London EC1M 5NP. Fax: 071 251 1442.

YES! Please keep me updated on the TopSpeed range of languages and compilers. I am interested in: C C++ Pascal Modula-2 Other _____

Name: _____ Position: _____

Organisation/Company: _____

Address: _____ Postcode: _____

Tel: _____ Postcode: _____

Jensen & Partners
International

EXE • 2-91

CIRCLE NO. 480

PANEL[®] Plus II



Screen Generator and User Interface Library for 'C'

PANEL Plus II features: Interactive screen design editor

- C and MS-FORTRAN code generators • library of user-interface functions • pop-up fields/windows • multi-line fields • mouse support • scrolling regions/scroll bars • help boxes • pull-down menus • custom validation • shadow borders • complete **library source**, with low-level source and headers for all supported systems (DOS, OS/2, Unix, VMS) • interfaces to popular PC **graphics libraries** • OS/2 DLL • **no royalties**.

Also available: **Utility Source Option** which allows **PANEL Plus II** to be ported to any suitable development platform. *New in version 2.1:* CUA/SAA compatible key assignments and edit functions, CUA 'function key area' support, word-wrap editing, new menu styles, **MS-C 6.0** interface (with on-line help), DOS-extender interfaces.

Panel Plus II licence fee per programmer: £295 (including library source and 700-page manual). With utility source option: £720. Please call for quantity pricing, full version list, and free demo disk.

Roundhill Computer Systems Limited

Orchard House, Ogbourne St. George, Marlborough, Wiltshire SN8 1SU
Telephone: 0672 84 535 • Fax: 0672 84 525

Roundhill also supplies compilers and development tools from Copia International, Graphic Software Systems, Lattice, Periscope, Pocket Soft, Polytron/Sage and others. Please call for a full catalogue and price list.

No excuse for user abuse

*Have you ever called in an HCI consultant?
John Harris explains what he does and why he is needed.*

I've just returned from INTERACT '90, a three-yearly international conference on HCI (human-computer interaction). A speaker, who used a PC and LCD transparency on the overhead projector instead of conventional slides, had us laughing at some classic design misconceptions.

When he'd finished with his first screen, he read out the last line *Hit any key to continue*. Marvelling at the choice before him, he slowly put his left hand to his mouth in exaggerated thought, as his right index finger hovered over the keyboard, finally pressing the top left-hand key. It was, of course, **Escape**... and the screen went blank. The audience erupted in laughter at such a graphic demonstration of a well-known design *faux pas*.

It's funny to see old chestnuts acted out so meticulously, but tragically they still occur in too many systems, wasting people's time, confusing them and causing them to make errors.

In this article I explore some of the things that go wrong and start to unravel what lies behind them. At least three communities are involved; developers, users and HCI experts. Although HCI experts may seem the most relevant, little can change unless developers and users play their part. I'll look at each in turn.

Software developers

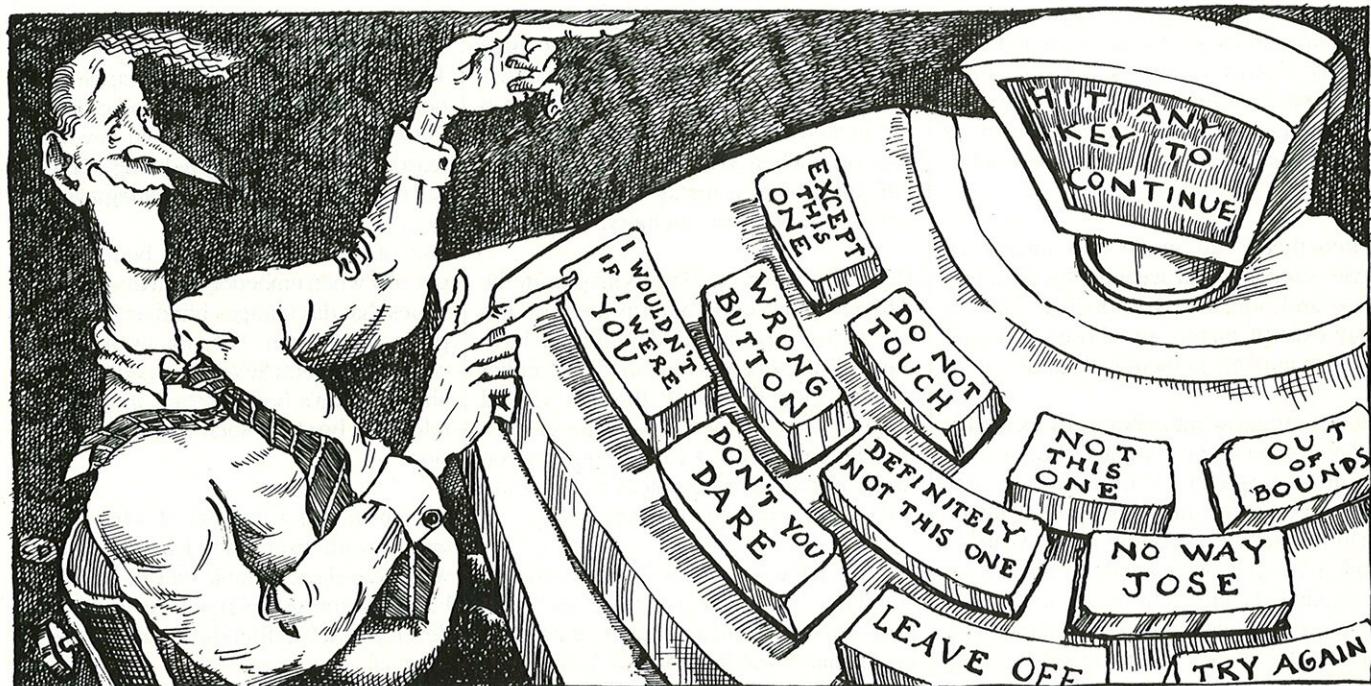
Developers tend to work with a systems approach. Priority is given to data and functions, and not to the user's-eye view, or how users' tasks are organised. This isn't surprising - computer professionals are seldom trained in human factors.

This 'computer-oriented' view shows up in the approach. In the early days, models of the development life-cycle were linear, and not amenable to techniques such as prototyping. Once a stage had been completed and signed off, you never went back to it.

More recently, structured methods and development models have become more tractable, allowing some degree of iterative design. But the malign influence of the past is still there.

Sometimes user interfaces are grafted on once the 'important' parts of the program are complete. The resulting problems sometimes lead to HCI consultants, like myself, being called in towards the end of a project. I have to tread carefully. Imagine the situation: you're on the point of completing a project, when an outside consultant is brought in to tell you how you've got it wrong!

There may be good reasons why things have been done the way they have, and I'm careful to get the confidence of the development staff. Equally they seek my confidence by showing they know about user interfaces. Sometimes they're amazingly well-informed. On other occasions, they



reveal that they're blindly following a guideline or quoting a saying, and they don't understand the issues.

Mythology trap

When guidelines are part of a thorough method they can be helpful. In designing user interfaces, however, mythology is as prevalent as methodology. It's a mythology that often lets designers down, and it's worth examining some examples.

Although I call it a mythology, its individual components - rules, guidelines and sayings - often contain more than a grain of truth. But as they get passed around out of context, becoming part of design folklore, they lose their *ifs* and *but*s, and get so distorted they finish up more of a hindrance than a help. A good name for such misleading guidelines would be *mythguidelines*, but I'll settle for misguidelines. Let's examine some misguidelines about menus, a very rich source of them.

A good example is *menus are userfriendly*. It crops up in various forms and places; in advertisements and sales literature for menu-driven software, and is quoted by people wanting to prove their awareness of human factors.

The reason it's misleading is because it begs important questions; blind application just results in a mess. The begged questions are about the type of user (new, occasional, frequent, typist, non-typist and so on), the type of menu, application, host system and so on.

The myth has sprung from the genuine observation that menus make options and responses explicit, lessening the need to look up command names, syntax and availability. However, in some forms it carries the implication that a system must be user-friendly just because it's menu driven, or that menus are always preferable to alternatives.

Often there's no single best interactive style; strengths and weaknesses may balance, and more than one adequate solution may exist. If menus are chosen, you still need to implement them with care.

Menus can grow into enormous hierarchical monstrosities on complex systems. This is most apparent when they're implemented as one-menu-per-screen, rather than pull-down or pop-up ones. As more and more functionality and/or data classes are added, the hierarchy grows and grows. To use a particular function or data set, you may have to go via several menus, only to

come all the way back immediately afterwards.

I met an extreme example in a financial dealing system. Users' actions changed the data it held, such as current prices, or who owned what. The user interface made it more like editing a database than dealing: simple deals involved convoluted sequences of actions, as you scaled up and down menus fetching records to edit and functions to edit them with.

Designers can provide short cuts, producing a network of menus rather than a hierarchy, but this adds to the complexity - it becomes more difficult to learn the system and easier to get lost in it. In most cases, it's better to structure the interface around the tasks the user wants to perform on the system rather than the functions or data available on the computer. Once the user has selected a particular task, the computer should make easily available the functions and data needed to perform it.

Seven Up

You can cut down the number of menus by having more options per menu. This is the subject of another misguideline, *don't have more than about seven items in a single menu* (the magic number quoted varies). The genuine origin of such a restriction is the limited capacity of human short-term memory. When you're making new and difficult choices you may need to hold the options in short-term memory.

However, experienced users may know the different options (long-term memory has enormous capacity) and prefer to have them all presented simultaneously, rather than picking their way through several menus. Even for new users, the choices may be easy, so that each option can be firmly rejected (or accepted) in turn, and not held in short-term memory.

Besides, if the seven item limit was true for computers, it would be true elsewhere - have you ever heard a restaurateur explaining the menu was kept short to avoid over-

loading customers' short-term memories? Restaurant menus are organised into courses, and meat dishes may be separated from fish ones and so on. We soon learn the conventions, and don't look for a chocolate pudding higher than the meat course.

It's equally important to organise big menus on computers. Palettes in a drawing or painting program can be easy to navigate - even when they're very large, such as on some CAD systems. These menus are set up in various ways, such as separating tool-selection items from pattern-selection ones, and similar patterns are usually grouped together.

There are even misguidelines on constructing menus. One is based on the assumption people search lists using the same 'algorithm' as simple programs. A system designer, embarrassed that he'd failed to observe the not-more-than-seven-options rule, proudly told me how he'd dealt with nearly 30 items in his pop-up menu. He'd put the most frequently used at the top. I gently pointed out that, with menus of that length, users try to learn the positions of the options and wouldn't always search from the top. The best way to help them is to structure the menu, putting together the six options on graphics editing, the five dealing with text labels and so on separating groups with lines.

Good structuring of menus not only speeds up searching and learning the positions of frequent commands, but helps you predict roughly where to find commands you've never used.

There's a misguideline which says that you should cater for all possible menu choices (itself a laudable objective) - *even if it creates overlapping choices between menus*. The flawed reasoning is: overlapping options doesn't matter, as both routes lead to the same place in the end. But new users reasonably assume they're being offered a real choice, and worry about getting it wrong.

Lists of good guidelines can be useful, especially when embedded in a discussion of issues that discourages blind and inappropriate application. See, for example, the chapter on Menu Selection Systems in Ben Shneiderman's book (details given in the reference box) for such a list and discussion.

I've given you a glimpse of what can go wrong with user interface design even when developers think they're observing human factors guidelines. If I'm to have any developers left as friends I'd better move on to users!

"AS YOU LIKE IT!"

Copy Protect AND/OR
Copy Register...Don't give it
away & cry!

EVERLOCK:
Stops copying & controls
programs as you want it.



EVERTRACK:
Does not copy protect
but it does control.

EVERLOCK

● COPY PROTECTS YOUR PRODUCT

● POSITIVELY SECURE AGAINST
BIT-COPY SOFTWARE

● PREVENTS DE-BUG, TRACING & DISASSEMBLY

● SUPPORTS ALL HARD & FLOPPY DISC FORMATS

● COMPATIBLE WITH MOST NETWORKS

● OPTIONAL RESTRICTION OF NETWORK USAGE

● LIMITS PROGRAM USE BY EITHER NUMBER OF
DAYS AND / OR EXECUTIONS

● REMOTE RESET OF INSTALL, DATE, EXECUTIONS

● SUPPLY UP TO 99 INSTALLS ON ONE DISC

● NO DAMAGED MEDIA OR NON STANDARD
DOS' ACCESS

● SUBSTITUTE I/O PLUG FOR SOFTWARE

● END USER TRANSPARENT

● EASY PRODUCT UPGRADE

● "SECURE STRING".
EG. USER REGISTRATION / SERIALISATION

EVERLOCK
NEW RELEASE
TAKE MORE
POWER

DUPLICATION
SERVICE

ANY
QUANTITIES
LARGE
OR SMALL
WITH OR
WITHOUT
PROTECTION

HIGH
QUALITY
STATE OF
THE ART
EQUIPMENT

TURNKEY
PRODUCTION
AVAILABLE

AS YOU
WANT IT!

EVERTRAK

● NON-COPY PROTECTION OF SOFTWARE
AUTOMATIC PRODUCT SERIALISATION / REGISTRATION

● ANTI - REVERSE ENGINEERING

● OPTION TO ATTACH PASSWORD

● LIMIT NETWORK CONCURRENCY

● DYNAMIC PROGRAM COMPRESSION

● DYNAMIC PROGRAM ENCRYPTION

● END USER TRANSPARENT

● COMBINE EVERLOCK & EVERTRAK

EVERTRAK
NEW RELEASE
TAKE MORE
POWER

Tel: 0527 585 550
Fax: 0527 65111

ICS House,
Edward Street,
Redditch,
Worcestershire,
B97 6HR. England.



CIRCLE NO. 482

TopSpeed to CeBit'91 with JPI

Occupying a site the size of a small town and spread over 23 halls each focussing on some particular aspect of computing and IT, CeBit this year runs for 7 days from March 13th to the 20th. JPI, (Jensen & Partners International), in association with .EXE magazine, is offering you the chance to win a trip to this spectacular event.

CeBit, always the scene for new product launches, will this year set the scene for the launch of TopSpeed version 3.0, a significant milestone in PC applications development. JPI, producers of TopSpeed C and TopSpeed Modula-2, will be showing version 3.0 of the TopSpeed integrated multi-language development environment. As well as being able to mix and match C & Modula-2 coded, JPI will add TopSpeed Pascal, (an ISO compliant Pascal with Objects and JPI extensions), and TopSpeed C++, (AT&T2.1).

The two first prize winners will each receive a ticket for their flight to Hanover. A limousine will take you direct to the fair for a full days browsing all that's new in IT. After a preview of TopSpeed version 3.0, you will join Niels Jensen, Chairman of JPI for dinner. On return to the UK you will receive a complete TopSpeed Professional Developers Kit consisting of the TopSpeed integrated multi-language programming environment and all four TopSpeed languages. Winners may choose either TopSpeed for DOS or OS/2.

Ten runners up will each receive TopSpeed environments and their choice of either TopSpeed C++ or TopSpeed Pascal for either DOS or OS/2. A further 15 runners up will receive a .EXE T-Shirt of their choice and a .EXE mug.

Simply answer the five following questions and send your answers by post to the address shown alongside:

KIBWORTH COMPUTER TRAINING

The rational choice for tuition in

C and C++

Our courses are an extremely rewarding experience. They are:

- * Conducted in really tiny groups to achieve proven and incomparable effectiveness of one-to-one dialogue.
- * Adapted wherever necessary to your particular needs. So no two courses are precisely alike, and they all teach as much as you can possibly learn in the available time.
- * Devised, presented, and kept up-to-date by the same vastly experienced programmer/tutor who can assess your needs.
- * Superbly equipped, and generously provided in every way.
- * Offered at competitive prices.

For details contact Eric Richards

Kibworth Computer Training
68 Springfield Crescent
Kibworth Beauchamp
Leicester LE8 0LH

0533 - 792653

CIRCLE NO. 483

COMPETITION

Q1: Which is the only ANSI compliant C compiler for DOS based applications?

Q2: Which Modula-2 shares common pragma directives with three other TopSpeed languages?

Q3: What two languages are being added to the TopSpeed family of integrated compilers?

Q4: Which compilers are brewed in England by Danes?

Q5: Which programming and systems journal celebrates its 5th year in (February) '91?

Name: Job Title:

Organisation/company:

Primary language used:

Compiler:

Operating System:

Address:

Phone: Fax:

Winners will be drawn from correct entries received by March 10th. Flight winners will be notified by fax the same day. All other prize winners will be notified by post. The judges decision is final:- Post or fax your answers to TopSpeed to CeBit'91. 3 The Mansards, Tavistock Street, Bedford, MK40 2RX. Fax +44 (0)234 217094

Users

It's their own fault, really. Users are insufficiently demanding of good user interfaces. This is partly because people are adaptable and can cope with poor designs, even if at great cost in errors, stress, speed of work and training.

Amazingly often I find these costs go unnoticed, or it's assumed they're inevitably part of having a computer system. Rarely are there arrangements for spotting user problems. End-users with ideas for improvements seldom find ways of passing on the information through supervisors to decision-taking management.

A new client of mine did have regular meetings between senior end-users, management and software staff. Over the years many extra facilities and improvements had been provided in this way, mostly successfully. But even here there were problems. In one case, the user interface for a new facility clashed with the original design - causing extra user errors. Although it's vital to involve end-users in the design process, they're no experts on user-interface design.

More typically, end-users accept what they're given, blaming themselves for any difficulties. This is very common, and it's worth considering an example in detail.

A user was asked to demonstrate a sales enquiry system. Suddenly he swore, and gave a embarrassed apology that he'd made a mistake. I asked him to explain. 'It's just a silly slip I often make,' he said. 'All the more important to understand why', I replied.

He wanted to abandon a line of enquiry and return to the base screen. His error was to enter *A* (for *Abandon* this line of enquiry) instead of *R* (for *Return* to enquiry system). From all other lines of enquiry *Abandon* would have been correct, but he was using a screen not originally designed (but now often used) for such enquiries and here *A* meant *Abort*, causing him to lose even the customer's name and the nature of the enquiry.

The next six users I spoke to all admitted making the same mistake and blamed themselves for doing something they knew was wrong. They were told during training not to enter *A* at this point; asking customers to repeat their name and enquiry isn't good for customer relations. The operators didn't suspect the error was connected with their high level of skill.

As users become skilled, lower level actions become automatic subroutines executed

with a single initiating thought or command. A skilled driver has only to think 'change gear'. A L-plated novice needs to think 'find clutch with left foot, press it down, move hand to gear-stick...'. Automatic, skilled actions are smoother, quicker, and leave you free to think about higher levels of your task.

With all this help and information available, there is no excuse for user abuse

Skilled users of the enquiry system think 'Back to base screen' and an automatic subroutine runs off the habitual low-level actions, entering *A*. The fact that the same high-level action occasionally requires a different low-level one, entering *R*, is an inconsistency that's a trap for the skilled. Either the advantages of skill are lost, as users concentrate on the details of the task, or they revert to habit at the risk of errors.

Abandon or Abort?

Investigating this led me to an identical result from a quite different cause. When the mainframe was busy and its response became slow, staff often typed ahead. On occasions, they would correctly enter *A* to abandon back to the base screen, but the computer wouldn't respond immediately. Assuming they had forgotten to enter *A*, they'd do it again. The computer would then use the first *A* to return to the base screen and the second *A* to abort - with the same consequences as before.

The system had many small traps like this, and a few more major problems. Senior staff had noticed a steady increase in the time it took to train new staff, but had no idea why.

What can we learn from this anecdote? First, users tend to blame themselves for making errors - and other difficulties - when much of the blame lies with the user-interface design. They're more likely to cover up errors than report them or discuss them with other users. Each user thinks that he's the only one having the problem, and nothing gets changed. So managers often don't know their staff are using a system that

could be far better - with higher rates of work, fewer errors and users getting greater satisfaction from their work.

Second, it shows how system use can change over time, especially when new facilities are added. When new uses are made of existing facilities, the original design may be violated and new inconsistencies can interfere with the skills in unexpected and unnoticed ways.

Third, problems with slow responses go beyond time wasted waiting. Experienced and time-pressured users will try typing ahead and, with no indication of the computer's response, they may misinterpret what has happened and make inappropriate entries.

Slow responses are something users *do* notice, discuss and complain about. The enquiry system staff had done just that. So the computer department logged response times. The printouts showed responses rarely exceeded a few seconds, even during busy periods. The perception of slow responses didn't match the objective measurements and I was asked to investigate. One reason was easily found. To navigate across several screens, each step or scroll involved a transaction with the computer. Subjectively, the response took *several times* a few seconds - very noticeable when you're keeping a customer waiting.

So the users blamed the computer for slow responses, but themselves for errors. If they hadn't complained about the response times, I'd never have been called in, and the other problems would still be concealed. If users do become more demanding, where can they and the developers turn for help?

The Experts

Certainly there is help available, but having taken both developers and users to task, it's only fair to put user-interface experts under the critical spotlight.

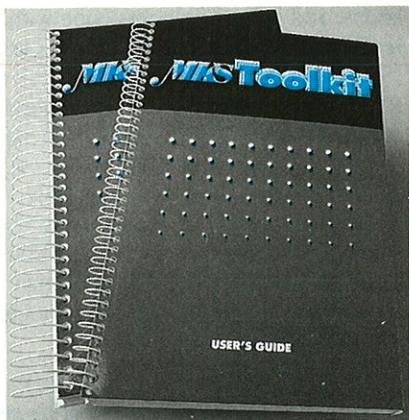
First, there are the would-be experts and not so experts. For example, people knowing little about the relevant subjects throw around the term *user friendly* to satisfy a client or beef up an advertisement, and now it's almost a term of abuse.

Potential experts - psychologists and ergonomics experts - were slow to apply their knowledge and skills to software. Even now, many experts are research-oriented rather than application-minded. Despite such research efforts, methods for designing user interfaces have been slow to appear, remain few in number and aren't widely

"The MKS Toolkit is an amazingly faithful replication of a System V UNIX™ environment."

- UNIX Review

Essential UNIX Tools for any PC user



"The entire MKS documentation package proved excellent in every respect"

Daniel McAuliffe, IEEE Computer, Jan. 1990

We can tell you that MKS Toolkit offers both experts and novices the purest form of UNIX utilities in a DOS or OS/2 environment. Fortunately, we don't have to. Software reviewers, universities and major corporations all over the world are discovering MKS Software and how it can help their programmers and software developers make the most of their creative talents.

With our proprietary code, MKS Toolkit offers you more than 150 UNIX System V, Release 4 compatible tools for DOS or OS/2. With MKS Toolkit, your computer or clone becomes a comfortable environment for shells, string matching, editing, file manipulation, and more.

In this one package you will receive a complete:

- MKS Vi full – screen editor
- MKS Korn Shell – user programmable interface

- MKS AWK – programming and prototyping language
- profiler – to analyse the execution time of code
- egrep/fgrep – fast pattern matching
- archivers, sort/merge, encryption, data compression & much more

Over 150 commands that help you get the most productivity out of your PC.

Easy Installation

With MKS's menu-driven installation you can be up and running in less than 10 minutes, no matter what your level of computer expertise.

"MKS software is absolutely the best in its class. Don't mix environments without it."

Grover Righter, Novell

Interconnectivity & Lan Licences

MKS Toolkit interacts well on most standard PC and PS/2 networks. Combined with Novell Netware™, the most popular LAN for PC's, MKS Toolkit creates a UNIX time sharing system in DOS or OS/2 organizations. UNIX workgroups can now hook up all of their PCs using PC-NFS™ and MKS Toolkit, enabling you to use a PC as a UNIX workstation and off-load your mini or main machine.

Spanning Novice to Expert

This versatile software reflects its users' needs. UNIX experts will have all the familiar power they demand while novices will find our documentation filled with excellent tutorial material.

Organizations such as AT&T, Hewlett-Packard, ITT, and NCR – all heavily committed to the UNIX system –

use MKS Toolkit to create a standard operating environment. Universities from King's College to the University of Berlin use MKS Toolkit to enrich personal research computing environments and double the potential of their PC teaching labs.

"With this package, you can become familiar with the UNIX environment on your microcomputer, with DOS only a keystroke away."

Byte Magazine, May 1989

Cost-Effective Learning Tool

If your organization is committed to moving into the UNIX environment, then MKS Toolkit is the perfect learning path. DOS or OS/2 users retain the familiar world of their PC keyboard and programs yet can move to a desktop UNIX environment at a keystroke. Exposure to new commands and functionality now becomes an integral part of the novice's working day.

MKS Programming Platform

MKS Toolkit can be purchased as an individual product or you can buy MKS Programming Platform which contains MKS Toolkit and three other MKS products. MKS Platform helps smooth out the details of programming and software development by adding time-saving utilities such as: MKS RCS (Revision Control system), MKS Make (an automated program builder), MKS LEX & YACC (compiler learning and construction tools).

In all, you simply cannot find a more complete set of commands and utilities to get from DOS or OS/2 to UNIX. Once you have used MKS Toolkit, programming will never be the same again.

US List Price

MKS Toolkit (DOS & OS/2)	\$399.00
MKS Toolkit (DOS/2 only)	\$349.00
MKS Toolkit (DOS only)	\$249.00

Please contact MKS dealers for UK domestic prices.

The Software Construction Co. Ltd. (0763) 244114; fax (0763) 244025
Grey Matter (0364) 53499
Systems Science (071) 833-1022; fax (071) 837-6411

35 King Street North, Waterloo, Ontario, Canada N2J 2W9

MKS, MKS Make, MKS Toolkit Korn Shell, and MS RCS are trademarks of Mortice Kern Systems Inc. UNIX is a trademark of AT&T.
PC-NFS is a trademark of Sun Microsystems.

MKS

Put The Power Of UNIX In Your PC.

known. Design tools for user interfaces were also slow to appear and are mainly confined to providing pre-programmed elements.

Sometimes the misguidelines that mislead designers have originated from research into human-computer interaction or psychology. One reason was over-interpretation of results of early experiments. Often later experiments produced apparently conflicting results when the context or measures were changed.

Some people assume that research on HCI is all about new methods of I/O. This is far from the case. There's still much to find out about basic topics, such as choosing command names, itself an area of apparently conflicting results.

For example, should command names be natural phrases, computer-oriented ones, or new words made up specially for the purpose? This whole area is much more complicated than even the researchers first thought. A review by Barnard and Grudin in the *Handbook of Human-Computer Interaction* shows the complexity behind what appear to be simple choices, but they also manage to dole out some sound advice

based on our rather limited state of knowledge.

Users blamed the computer for slow responses, but themselves for errors

This can be repeated for many other topics; even the specialists don't have all the answers, but in most cases they do have *some* answers and can often give very relevant advice about how to proceed in difficult areas.

As important is the general approach taken by people whose first responsibilities are to understand the users and their tasks, and to act as their advocates in the design process. Some methods (eg task analysis) and tools (eg for prototyping) are already available.

What can be done?

If progress is to be made, users will have to demand more usable systems. Developers, even the readers of .EXE, can't be expected to spend the time and effort needed if there's no demand. Users are often unaware of the problems, of their cost, of what can be done about them and of the availability of specialists.

Even the government seems worried. Instead of expecting the market to sort things out, it's putting DTI money into an HCI-awareness programme called *Usability Now*. Launched in November 1990, it includes guides, case studies, a newsletter, help with access to consultants and more. It aims to make users aware of usability issues, and help developers cope with the resulting demand for more usable systems.

User organisations will be encouraged to make sure developers have relevant expertise (eg for graphical user interfaces). As developers you can seek specialist help where necessary - but you too must check experts' competence. The Design Council publishes a listing of approved human factors consultants (reference box), though it isn't comprehensive.

F77L-EM/32

Port 4GB mainframe applications to 80386s with this 32-bit DOS-Extender compiler. The winner of PC Magazines's 1988 Technical Excellence Award just got better. New Version 3.0 and OS include: Editor, Make Utility, Virtual Memory Support, DESQview Support, New Documentation and Free Unlimited Runtime Licenses. F77L-EM/32 and OS/386. POA.

F77L

The compiler of choice among reviewers and professionals. Includes a Debugger, Editor, Profiler, Linker, Make Utility, Weitek and 386 Real-Mode Support, Graphics. POA.

Lahey Personal Fortran 77

New Version 3.0: Full ANSI 77, Debugger, Editor, Linker, Library Manager, Microsoft and Borland C interfaces, 400 page Manual. POA.



When people talk about FORTRAN the name mentioned most often is **Lahey**
Computer Systems Inc.

System Science

CIRCLE NO. 486

3-5 Cynthia Street,
London N1 9JF
Tel: (071) 833 1022
Fax: 081 837 6411

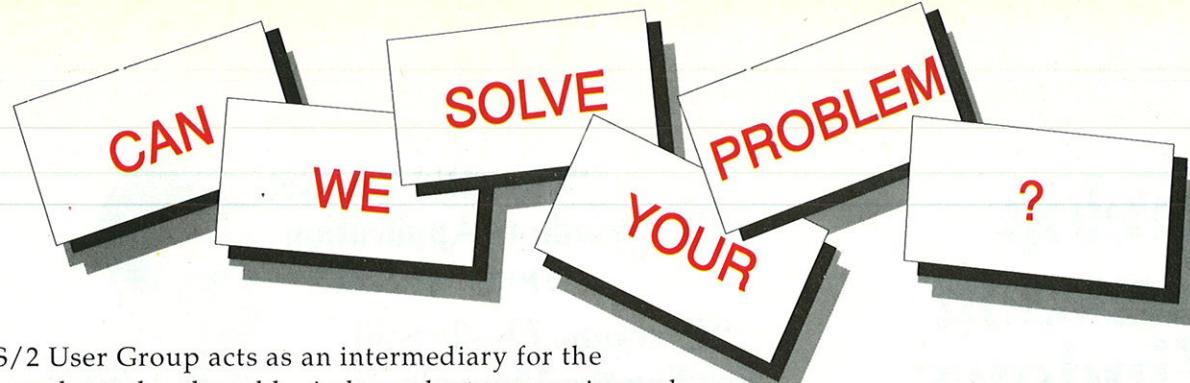
KJD
Software Systems

£200 + VAT & Carriage
Please telephone for details:
(0424) 721761



Unit 10, Mercatoria Business Centre,
100-102 Norman Road,
St Leonards-on-Sea, E.Sussex. TN38 0EB

CIRCLE NO. 487



The OS/2 User Group acts as an intermediary for the sale of products developed by independent companies and programmers. There are currently eight software products on sale with substantial discounts to members:-



The Business Graphics Library

A collection of functions in a dynamic link library which may be used in OS/2 Presentation Manager applications to display information pictorially. The library includes a total of over 20 functions including bar charts, line graphs, pie charts, text output, printer, thick line and attribute functions. All the charts will accept any reasonable number of values, and are device independent so may be used to display on a window or print on a printer. The library is supplied as a DLL file together with the header and LIB files required to use it in a C program, a demonstration program, and the source of an example program together with full documentation.



REMDEB

REMDEB is a remote debugger for OS/2 which makes debugging PM programs easier. It can also be used for full-screen OS/2 programs. It works by running REMDEB on one machine and your usual debugger on another. It runs across a LAN thus avoiding the transfer bottle-neck of a serial line.



Dualboot

The DUALBOOT system provides a dual boot mechanism for IBM OS/2 1.1, either Standard or Extended Edition, and PC-DOS 3.3. This can provide the user with a choice of which operating system to boot. However, unlike other dual boot systems, DUALBOOT allows the user to instruct the system (by running a utility) to boot a chosen operating system all the time. The booted operating system can be changed simply, again by running a utility, or the normal option operation can be reverted to in the same way.



Disk Daemon/2

Disk Daemon/2 is a new and powerful OS/2 full screen sector and file editor for the PC and PS/2 family. It allows the user to view or edit both absolute sectors and the contents of PC files with up to 3 files being opened for editing at the same time.

NAME: _____
ADDRESS: _____

TEL NO: _____

MEMBERSHIP NO: _____

Cheque enclosed for £ _____ +VAT

ACCESS VISA EXPIRY DATE

Please send further information about membership.



Snoop/2

Snoop/2 allows the developer to view the status of various internal OS/2 structures. It uses a device driver to acquire its knowledge, and can therefore be used on any OS/2 system. It also allows you to view and edit any location in memory, while running programs and dynamic link libraries. (Family Application).



Disa

A User-friendly dis-assembler that allows the developer to view their programs after they have been compiled and linked. Files supported are: .exe, .dll, and .sys files. (Family Application).



Kill-By-Name

A kill-by-name program allowing you to stop - kill - programs by name. (Family Application).



Harness

Harness enables you to debug device drivers using Codeview. It allows you to single step and set break points inside installed device drivers. Initially written as a DOS application, Harness also works with OS/2 installed device drivers.

User Group Members

Non-Members

Business Graphics

Library	£75.00	£95.00
REMDEB	£75.00	£95.00
Dualboot	£30.00	£50.00
Disk Daemon/2	£30.00	£50.00
Snoop/2	£150.00	£190.00
Disa	£150.00	£190.00
Kill-By-Name	£40.00	£60.00
Harness	£75.00	£85.00

NUMBER REQUIRED	PRODUCT	PRICE INC. VAT



International OS/2 User Group, Cecily Hill Castle, Cirencester,
Gloucestershire. GL7 2EF England.

Telephone : (Intl +44) 0285 655888 Fax: (Intl +44) 0285 650537

Develop Microsoft Windows Applications Fast!



"How Did You Get
This Windows Application
Done So Fast?

"Everyone Else Is Still
Reading the Manuals..."

With full VERSION 3.0
Windows Support

Two New Products
For C And Actor Programmers

WinTrieve™
ISAM indexed file manager. Only £295, no royalties.

Whitewater Resource Toolkit™
Edit dialog boxes, bitmaps, icons and more. Only £145.

Call Now For More Information!

The Windows Advantage

You've selected Microsoft Windows as your graphical user interface. Windows applications are easier for end-users to learn and use. It is the standard for PCs.

Now you want Windows applications completed as fast as possible. But it takes time for your programmers to get up to speed on the intricacies of Windows development.

Why The Delay For Windows Applications?

Programming for the Windows environment is very different from character-based programming. Just learning the 450 Windows function calls can take months. Getting used to its event-driven system and memory management takes additional time.

Programmers who are new to Windows development are looking for support to help them produce high-quality Windows programs quickly and efficiently.

3 Ways To Speed Up Windows Application Development

Now there are specific ways to speed up applications development under Windows.

- **Code reduction.** There is a fast, effective way to develop Windows applications. A way that cuts the amount of

code needed. A way that by-passes much of the complexity of Windows programming. A way that doesn't compromise performance.

- **Short learning curve.** There is an easy, convenient way to get your entire programming staff up to speed with Windows within weeks.
- **From prototype to complete application.** There is a way to quickly develop prototypes of Windows applications in days, not months. Then, turn those into completed applications quickly.

1 Product, 3 Solutions

There is one product which brings all three solutions to your Windows programmers. It's called **Actor**®.

Actor is a proven product. It's now being used in thousands of software development groups world-wide.

Actor Does Windows

Actor gives your programmers a fast, effective way to develop Windows applications. **Actor**'s complete development environment allows you to produce everything from prototypes through stand-alone applications. Programs developed in **Actor** are fast and support all Windows features.

Actor offers interactive source code debugging and testing—greatly reducing development time. Your programmers

benefit from high code reusability because **Actor** is an object-oriented programming language. This allows them to build on existing code, eliminating duplication of effort. They can even dynamically link to existing C code.

Actor also provides one more essential: fast, responsive support from The Whitewater Group®, recognized worldwide as leaders in Windows programming.

If you are programming applications for Windows, you should be using **Actor**. Users report that **Actor** cuts Windows programming time by at least 50%.

Act Now!

See for yourself how **Actor** can help your Windows programmers! The first step is to order our free booklet, **For Faster, Better, Smaller, Smarter Windows Programming**. Examine detailed information about using **Actor**, complete with actual source code examples.

Call us at 0628-665722. We'll send you the booklet and complete details about **Actor**.

Take 60 seconds now to save months of development time in the future.

NEOW - The Windows Specialists

Unit 12, Progress Business Centre,
Whittle Parkway, Slough, SL1 6DQ.
Telephone: 06286 68334. Fax: 06286 65722

All trademarks acknowledged

Don't feel threatened by more knowledgeable users. Many development problems occur because of a lack of understanding between users and developers. It's important to involve users, including end-users, in many aspects of development. Investigate user issues with them at the requirements stage, not later.

Prototype the user interface and let end-users try it out.

It's helpful if your users understand the stages of development, their own roles and those of the developers. Refer them to *Developing Systems Together* (reference box), a booklet that explains all these in non-

technical language with the aid of cartoons; it even has a short chapter on the user interface.

If you want to learn more yourself, there are several channels. Consider possibilities for training in HCI; for example, there's a new Open University course. There are some good and readable HCI textbooks - but be careful not to get bogged down in the large 'researchy' tomes written for specialists by fellow researchers.

For specialists there are other information sources, such as the HCI literature service run by the Scottish HCI Centre. The HCI Specialist Group of the British Computer Society offers meetings and a newsletter.

With all this help and information available, there is no excuse for user abuse!

EXE

Dr John E Harris runs HCI consultancy, CAMBIT The User Interface (0223 811487). He is a Chartered Psychologist with a background of research in cognitive psychology at the Medical Research Council, two years with Scicon, followed by Alvey Project work and user documentation.

"How Did We Ever Manage Without SMS?"

ASM (Europe)

SMS provides a comprehensive set of tools for automating the process of managing different versions of files and configurations of systems. SMS improves quality, reliability and productivity.

Version 2.1 now available! Even more power and flexibility

Software Management System provides a fully integrated environment; all features can be accessed through a menu-driven Front-end with context-sensitive on-line help.



Version Control

Management of multiple revisions, lines of development and users

- Revision Storage and Retrieval
- Edit Control
- Revision Identification and Reporting
- Minimisation of Storage
- Text and Binary File Support

Configuration Management

Management of configurations of the items which make up a project/product

- Release and Configuration Identification
- Automated System Building
- Variant/Configuration Maintenance
- Automated Dependency Generation

Modification Requests

Formalisation and tracking of bug reporting and upgrade procedures

- Change Management
- User-definable Life-cycle
- Active Notification System
- Modification Analysis and Reporting

Available for: MS-DOS, OS/2, Unix, VMS, OS-9
 Single User PC Price £ 490.00 + VAT ; 5 User PC Price £ 980.00 + VAT
 Please contact us for further technical and pricing information

INTASOFT : the productivity tools for professional software developers

60 Portland Street, Exeter, EX1 2EQ, England, Tel: 0392 217670, Fax: 0392 412463

Meeting the deadlines

The real time executive is the heart of a fast real time system, and the subject of volumes of academic research. But Jim Cooling is not impressed.

Electronics has played a major part in the control, monitoring and instrumentation of 'real world' systems for many decades now (the Centurion tank gun stabiliser, for instance, first saw the light of day about 45 years ago). Three particular features were originally identified as essential requirements of these systems: reliability, safety and deterministic behaviour.

There was, in reality, a fourth requirement: speed of response. Yet for the analogue (and, later, the digital) designer, this was a non-issue. Electronic controls were so fast (compared with the system being controlled) that speed was not a problem. Another feature marked out these real world systems; the electronics formed an integral part of the **total** system. That is, their design and function didn't make sense taken in isolation from the application.

When the microprocessor arrived, circa 1970, it seemed (with its power, flexibility and programmability) like the ideal device for future systems... until we encountered the Von Neumann bottle-neck. Suddenly response times **were** an issue - and an important one at that. It was clear that real time systems had definite, often rigid, deadlines. Failure to meet these produced unac-

ceptable - sometimes catastrophic - results. Computing speeds, it turned out, were a crucial factor in 'real world' applications.

lines without causing problems. This is most definitely not true of hard applications.

All the work was done on a VAX 11/784. Very useful for people working with microcontrollers

By the late 1980s, conventional electronics had, in the main, been replaced by computer techniques. To distinguish these reliable, deterministic and timely systems from, say, on-line computing, a new name was coined: 'hard real time embedded systems'. Now, does this also imply that hard real time systems are fast? Many would say no. Their argument is that any system having *definite* deadlines falls into this category - *actual* response times are not important. Unfortunately, if this definition is used, the practical difficulties of meeting fast deadlines is often ignored (certainly true of much current research literature). I believe that real time systems should now be categorised into four groups, as shown in Figure 1.

The boundary between 'fast' and 'slow' is obviously subjective (here is a personal view: where response times are measured in seconds, systems are slow). In contrast, the distinction between soft and hard responses is very clear-cut. Soft systems - no matter how fast they are - can go late on their dead-

The fast response

How can you guarantee fast responses from computers? Probably the most widely used method is to make time-critical tasks interrupt-driven. Where a system contains only one such task this is very easy to implement - and be confident of meeting the deadline. However, few embedded systems are this simple; most consist of several time-critical processes. For example, the flying laboratory system fitted to a Canadian research aircraft¹ contains (among others) the software processes shown in Figure 2.

Moreover, it isn't unusual to find that while one task is being processed, others demand service. To control processor behaviour in these circumstances, the programmer normally assigns priorities to tasks (in the Figure 2 example, the ordering is by priority, with 1 being highest). In general, a running task may be replaced ('pre-empted') by a higher priority one which requires service (though tasks may be blocked out by disabling interrupts). The problem is: when using this technique, how can you be sure that all tasks will meet their deadlines? In small systems, tasks are visible enough for the programmer to estimate accurately system responsiveness. But what about large systems? And what about designs based on true multi-tasking software, running under the control of a real time executive?

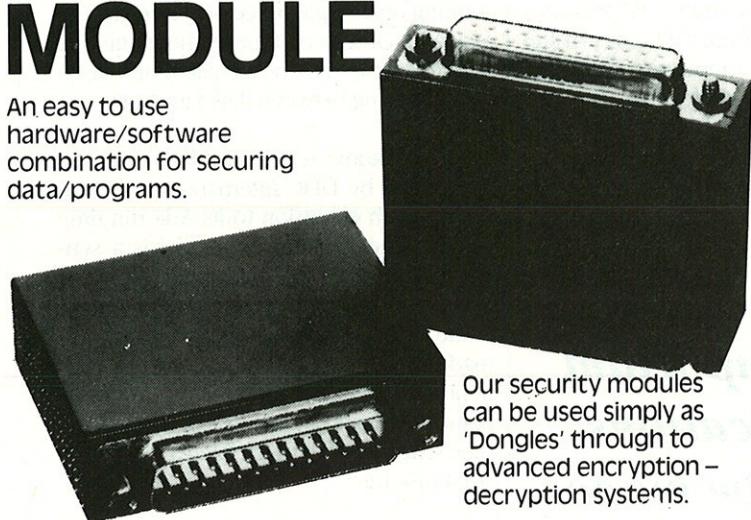
Modern executives designed for real time embedded systems (such as Ready Systems VRTX32 and Software Components Group pSOS) use priority task scheduling schemes. The difficulty here is to decide on the 'best' way of setting schedules. A very common approach is to assign priorities at compile time, leaving them fixed thereafter (a static schedule). Priorities may also be

	SLOW	FAST
SOFT	ON-LINE INTERACTIVE SYSTEMS	CAE GRAPHICS SYSTEM
HARD	SLOW CHEMICAL PROCESS PLANT	FLIGHT CONTROL COMPUTER

Figure 1 - Categorising Real-Time Systems

SOFTWARE SECURITY MODULE

An easy to use hardware/software combination for securing data/programs.



Our security modules can be used simply as 'Dongles' through to advanced encryption - decryption systems.

- ★ For IBM PC family (and compatibles)
- ★ Variants for other systems
- ★ Advanced construction - rugged and compact
- ★ Proven throughout the world
- ★ Transparent operation - user friendly
- ★ Driver software supplied for variety of applications



Control Telemetry of London
11 Canfield Place, London NW6 3BT
Tel: 071-328 1155 Fax: 071-328 9149

CIRCLE NO. 491

PROTECT YOUR PROGRAM FROM PREDATORS

- Established for over eight years in software security products
- Internationally established product
- Competitively priced keys (from £18.00)
- Supports IBM PC/XT/AT and PS/2
- Supports Apple Macintosh range
- Advanced manufacturing techniques
- Transparent and user friendly
- No drivers necessary (with exception of programmable key)
- Large range of security keys.

For further information and 60 day evaluation kit, contact:

COMSEC (U.K.) LTD.
Cranfield Airport,
Cranfield,
Bedfordshire,
MK43 0AL

Tel: (0234) 751203
FAX: (0234) 751082



COMSEC
(U.K.) LIMITED
COMPUTER &
COMMUNICATIONS
SECURITY SYSTEMS

CIRCLE NO. 492

System Science

Microsoft Languages

C Professional Development

The latest C 6.0 has improved optimisation, in-line assembler and draft ANSI support amongst other enhancements. The new Programmer's Workbench integrates the operation of the compiler, make, linker code browser and the new Microsoft C Advisor. CodeView 3.0 can reside in extended memory for DOS, Dos and OS/2 support is included for just £245.00

Basic Professional Development

New version 7.0 handles large applications with run-time overlay support, far strings and EMS4.0 support. New included toolboxes for menus and user interfaces, presentation graphics and maths. New ISAM for transaction processing and a data dictionary. The Microsoft Basic advisor is in the Extended Quick Basic environment. DOS and OS/2 ... £275.00

Windows Software Dev. Kit 3.0

Windows SDK provides a set of tools and documentation for developing applications for the new Windows 3.0, including a resource compiler, dialog editor and libraries for DDE (dynamic data exchange). Use it with Microsoft C 6.0 or 5.1 to put the most into your Windows app. £325.00
Windows 3.0 £89.00

Fortran-77 Compiler ver 5.0

Fortran 77 with VAX and proposed 8X extensions, the compiler supports large arrays, large data and large code and graphics primitives. Editor, linker and CodeView, the source code debugger, are included. For DOS and OS/2 £245.00

OS/2 PM Toolkit

For OS/2 ver 1.2, the PM toolkit contains extensive documentation on OS/2 and PM programming, with software tools, such as the Icon Editor, Dialog Box Editor and resource compiler, with the on-line Quick Help system. Use the toolkit with C, Fortran or Basic £325.00

Further Microsoft Products

Assembler 5.1 (DOS and OS/2).....	£95.00
Quick C ver 2.5	£65.00
Quick C with Assembler ver 2.51	£98.00
Quick Basic ver 4.5	£65.00
QuickPascal ver 1.0	£65.00
MS Cobol ver 3.0 (DOS and OS/2)	£475.00
Microsoft Programmers Library (CD-ROM)...	£275.00

3-5 Cynthia Street,
London N1-9JF

Fax: 071 837 6411

Prices are exclusive of VAT, but do include UK delivery.

Prices are subject to change and alteration.



071- 833 1022

CIRCLE NO. 493

changed using program instructions (dynamic scheduling); clearly this technique must be used with great care.

This scheduling policy has two major drawbacks when used in hard real time systems. First, the programmer is totally responsible for ensuring that deadlines are met. Second (and very much a consequence of the first), processor utilisation is often quite low. A most unsatisfactory state of affairs - but how can the situation be improved?

The Future

In the future, scheduling policies must be developed that eliminate the need for programmer intervention, improve processor utilisation and ensure that deadlines are met.

We are going to see both short-term and long-term solutions to these requirements. The long-term ones will be generated by the computer science community - although current research papers suggest that practical schemes will be a *long* time in coming. The realities of hard real time embedded systems - size, weight, environment and response times - don't seem to be well understood by academics.

For instance, a paper entitled *Implementation of a time-driven scheduler for real time operating systems* looks inspiring - until it turns out that all the work is done on a VAX 11/784. Very useful to people working with microcontrollers. And *Inprecise results: utilising partial computations in real time systems* turns out to have been developed on a set of networked Sun workstations. I can't quite see the European Fighter Aircraft hauling that lot about the sky, even if processors are becoming more powerful. A paper entitled *Implementation and evaluation of a time-driven scheduling processor* contains the following illuminating statement: 'For the loads used in our experiments, the average task processing time was 0.5 seconds per request. This was usually sufficient to avoid host processor idle periods, awaiting the outcome of occasional long scheduling decisions'. With response times like that on offer, you don't need a computer - you can do better all by yourself. Some good work is going on, but it's going to be quite some time before it becomes usable.

Short-term solutions will (and in some cases, already have) come from the suppliers of real time executives. Commercial pressures are forcing the pace, even if it's only a matter of keeping up with competitors. Emerging systems may not be perfect, but they do hold potential for the future. Two basic scheduling techniques are offered: 'rate monotonic' and 'deadline driven'. A combination of these, 'mixed' scheduling, is also likely to become another practical scheme.

scheduling. However, be warned; 'utilisation' is a misleading term. A set of tasks running on a single processor, using all the available processor time without missing deadlines, can have a calculated utilisation factor varying between 0.693 and 1.0.

Rate monotonic scheduling has been implemented by DDC International of Denmark³ as an extension to its Ada run-time system. The Ada tasking model is a synchronous one, so this scheduling scheme fits in naturally with Ada programming. Some important modifications had to be made to the technique to make it effective in practice (NB: these were needed to overcome drawbacks in Ada, not in rate monotonic scheduling!). The problems and DDC's solutions are:

- 1) Tasks waiting for a rendezvous are held in a FIFO queue. A low priority task delays a higher priority one further down the queue. *Solution: use priority based queues.*
- 2) During task rendezvous, low priority tasks can block the execution of higher priority ones ('priority inversion'). *Solution: use priority inheritance. If a task is blocked by a lower priority one, the blocking task inherits the priority of the blocked task.*
- 3) The Select statement selects executable tasks in a random fashion. No attention is paid to priorities. *Solution: Choose the highest priority task within the Select operation.*

DDC says, 'measurements have shown that processor utilisation can exceed 90% in some systems before critical deadlines are missed, where only 30% processor utilisation was achieved without rate monotonic scheduling'.

Even so, there are a number of major weaknesses with the rate monotonic method. First, it applies only to periodic tasks. As can be seen from Figure 2, real systems involve both periodic and aperiodic (asynchronous) processes. Second, it does not take task importance into account. In reality, a task at any repetition rate may require an urgent response once it is activated. Third, the definition and use of deadlines is not good enough for many applications. Provided a task finishes before it is next activated, it is considered to have met its timing requirements. Thus all the responses shown in Figure 3 are, by this definition, satisfactory. Yet the task elapsed time for task B is considerably different in Figures 3b and 3c. In many situations (for example, a marginally stable closed-loop control system), this variation would be unacceptable.

Some important modifications bad to be made to overcome drawbacks in Ada

Rate monotonic

Rate monotonic scheduling is simple and straightforward - but applies only to periodic ('synchronous') tasks. Here, tasks are assigned priorities which depend on their repetition rates (frequency). The highest frequency task gets highest priority, lower frequency ones having successively lower priorities. The one with the lowest repetition rate ends up with the lowest priority in the system. Tasks are then run under a pre-emptive scheduling policy, as shown in Figure 3.

Note that no attention is paid to task execution times when setting priorities. Note also that, as defined here, a task's deadline is the time interval between successive activations.

This technique was analysed in detail by Liu and Leyland² in 1973, who developed a measure called *processor utilisation* which is often quoted in relationship to rate monotonic scheduling. This is used to calculate whether tasks can be guaranteed to meet their deadlines when using rate monotonic

PROCESS NAME	RATE
1. Urgent command processor	Asynch
2. Magnetic field counter	8 Hz.
3. System process	2 Hz.
4. Loran_C receiver	0.1 Hz.
5. Pilot's command processor	Asynch

Figure 2 - Process Timing Example



There are many ways to boost your software development efforts

The most sensible way of launching your next software development project is with Enfin/2, a new 4th Generation Language software development tool which allows you to produce complex software applications quicker and more easily than you had previously thought possible.

It lets you create applications for either Microsoft Windows 3 or OS/2 Presentation Manager Graphical User Interface (GUI), with built-in graphics, charts, data storage, SQL database, sophisticated graphical reports and communication links to the outside world. Applications developed in one environment will run in the other.

Enfin/2 lets you build complex systems out of simple building blocks, or 'objects'. It supplies nearly 150 standard objects to give you a valuable head start, allowing you to prototype, explore your business requirements and come up with

solutions for your business' changing needs.

But things don't stop there.

A sophisticated range of code generating editors exists within Enfin/2 to increase productivity, whilst an optional compiler produces standalone, royalty-free applications to use on as many workstations as you like.

Enfin/2 is backed by QA, the UK's premier PC technical training and products company, who will train, consult and support your efforts. Contact us for more details, or to book onto one of our free introductory seminars.

... And do away with your box of matches.



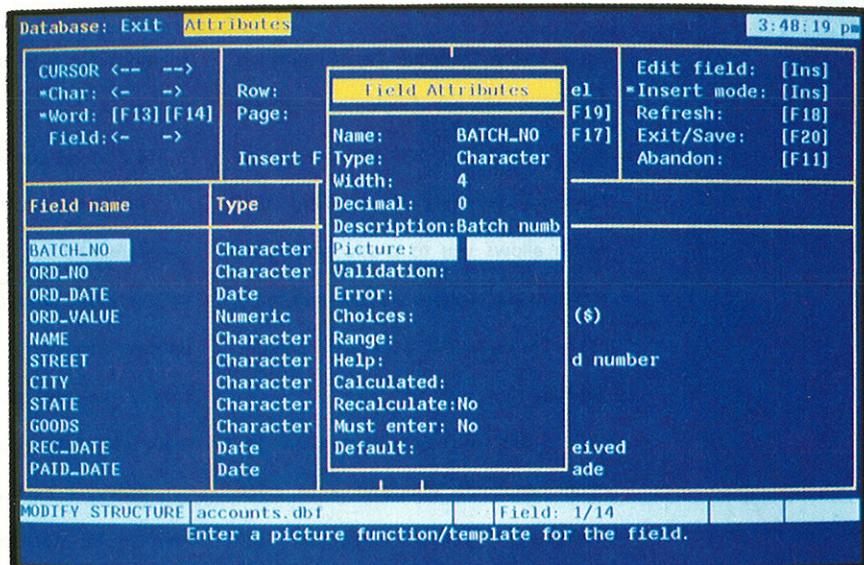
Tick here for full details: ENFIN/2 for OS/2 ENFIN/2 for Windows Introductory Seminars



NAME	TITLE
COMPANY	
ADDRESS	
POSTCODE	TEL. NO.

Mail to The Products Group, QA Training Ltd., Cecily Hill Castle, Cirencester, Gloucestershire GL7 2EF. UK.

dBase III Plus, FoxBASE & Clipper applications running on VAX & UNIX?



Try our 30-day software licence
In an hour or two you'll be converted

Can Recital really be that easy?

The answer is yes. And to prove it, we invite you to send for our full-featured 30-day software licence for VAX/VMS™ and UNIX systems. Recital is so compatible and easy to use, you'll have your dBase™ applications up and running in Recital in about an hour. Without any additional programming or modifications.

Then take the rest of the month to discover the richness and power of this 4GL system. Recital includes tools like a report writer, screen painter, transparent access to RMS files, an integral data dictionary, popup calendars, calculators, pick lists, note pads and much more. All on a VT-type terminal!

Your 30-day license will include the full Recital system, documentation and hotline support. How can you lose?

Order today by calling one of the numbers below. Or write to Recital Corporation.

Recital Corporation Limited
South Bank Technopark
90 London Road
London SE1 6LN, UK
Tel 071-401 2727
Fax 071-633 9617

Recital
CORPORATION

All trademarks are the properties of their respective companies

Recital Corporation Inc
85 Constitution Lane
Danvers
MA 01923, USA
Tel (508) 750 1066
Fax (508) 750 8097

Deadline scheduling

The limitations of rate monotonic methods led researchers to look at dynamic techniques for scheduling tasks. The most promising turned out to be what is called *deadline scheduling*. Again, Liu and Leyland had a big hand in this. However, they stuck with their original definition of 'deadline' which really wasn't satisfactory. A much more useful measure - and highly applicable to real time systems - is the permissible delay between invoking a task and having it completed. We'll call this the task *due-time*. Further, our definition of a task deadline now becomes 'the time at which the due-time expires'. Within this system the important timing functions are:

- Task execution time.
- Required response time.
- Activation time.
- Elapsed time from activation.
- Time to deadline.
- Amount of task execution currently completely.
- Spare processing time [that is, (e) minus (f)].

The purpose of modern deadline schedulers is to ensure that system tasks are com-

pleted by their due-times. To set a task's due-time, we have to allow for three factors:

- How quickly a task needs service.
- How critical a task is.
- The actual task execution time.

Note that task repetition rates don't enter into the equation.

Each task has an initial low-level priority value, which may be set at compile time. When a task is invoked, it initially runs at this value. However, as it gets closer to its due-time, the priority is raised, as shown in Figure 4. Once the task is completed, its reverts to its initial value.

When using a simple implementation of deadline scheduling, the task closest to its deadline has the highest priority. The one furthest away has the lowest. The task scheduler always sets the highest priority ready-task running at reschedule time.

At first sight this seems to have cracked it - but there is no such thing as a perfect solution. Even the simple approach described here has one major problem: time. Deadline scheduling is a complex task. Implementing it in software will add substantially to the time overhead of the executive. In a soft real time system there should be enough processor time to run the scheduler in software. But what about fast systems?

In my view, the only way to meet such requirements is to off-load the scheduler into silicon - ie use co-processing. This is how the PORTOS ('Portable Real Time Operating System') deadline driven scheduling system⁴ is built. With this system, timing factors are included in two ways. Items (a) and (b) noted above are defined at compile time. Items (c) to (g) are handled by an ASIC specifically designed for the PORTOS system.

At the present time, PORTOS is available for IBM PCs, single board computers (eg the Intel SBC 286/310) and embedded systems based on Intel microprocessors.

Problems to be solved

The DDC and PORTOS executives represent significant steps in the development of real time executives. But they don't have all the answers to the needs of multi-tasking for real world applications. The next stage would seem to be the combination of deadline and rate monotonic techniques - 'mixed scheduling'. Even here there are

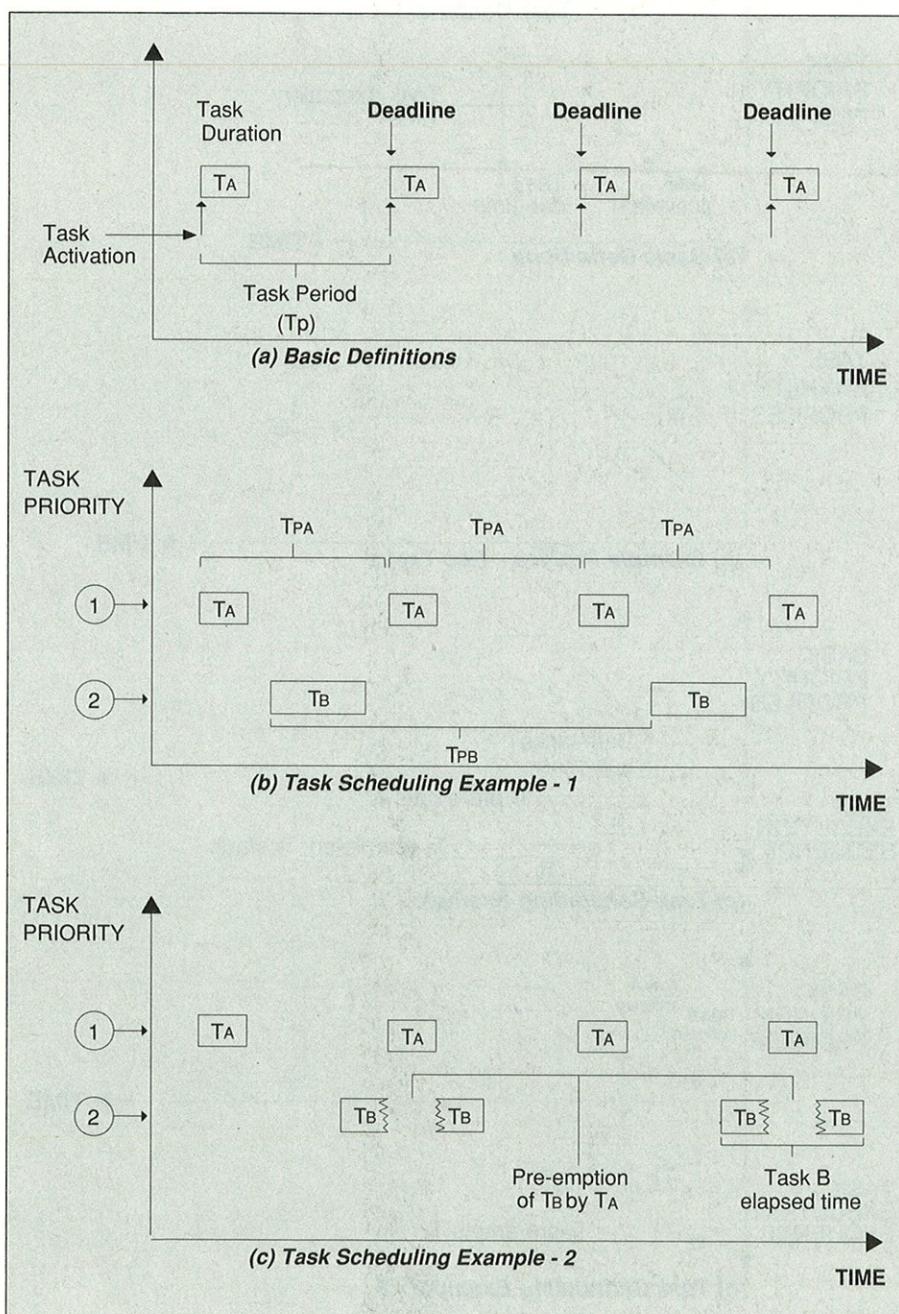


Figure 3 - Rate Monotonic Scheduling in Action

a number of thorny problems to be tackled, all of them inter-linked. The major ones are:

- (a) Defining the 'best' task priority profiles.
- (b) Linking criticality with profile.
- (c) Handling overloading.
- (d) Predicting performance.

Many priority profiles have been proposed for deadline scheduling; few have been implemented. There is really very little experience in this area. Further, complex profiles may prove to be great in theory and simulation - but hopelessly impractical for real time work.

A second issue is that of criticality. Suppose that serial data arriving in a UART must be collected within a set time period; otherwise the data will be lost or corrupted. Further suppose that a task is being executed which has a longer deadline than the UART task - only it must *not* be preempted for reasons of safety or security. How should this be handled?

This leads to the question of overloading, that is, tasks missing deadlines because of processor loading. Proposed solutions to this problem are based on the idea of executing critical tasks (which can then be guaranteed to meet their deadlines) and dropping or deferring others - *load shedding*. It is not currently clear how can we implement these effectively and safely, **and** within the time and performance constraints of real time systems?

Finally, a method is required for predicting the performance of application programs at the design stage. And, even more important, a technique is needed to measure and record the *actual* performance of the software within the target system.

Multi-processors

Operating systems for hard real time multi-processor systems are in their infancy. The overall subject is currently a 'hot' one in the research community. Unfortunately, much of the work seems far removed from the world of real systems; surveying the research publications of the last few years only induces great gloom. Not only is much of it verbose, pedantic and verging on the incomprehensible; it also frequently fails to address some really important questions. For instance, many researchers are heavily into 'load balancing' - the concept of shifting tasks from processor to processor to meet scheduling requirements. Load bal-

ancing algorithm development is top of the agenda. Yet questions such as:

- How do we develop the software for such schemes?
- How do we keep track of software execution?
- Do we have the time to evaluate loading and redistribute tasks?
- Do we even have the necessary information to do this?
- How do we ensure correct and timely inter-processor communication?

- How do we handle and recover from communication errors?

are rarely raised. We have a long way to go before the multi-processor problem is solved.

Conclusion

It seems clear that, in the near future, scheduling methods for hard real time systems will generally remain unchanged. Current fixed/dynamic priority based techniques will continue to be widely used. However, rate monotonic and deadline scheduling systems will be gradually introduced, together with combinations of these. But

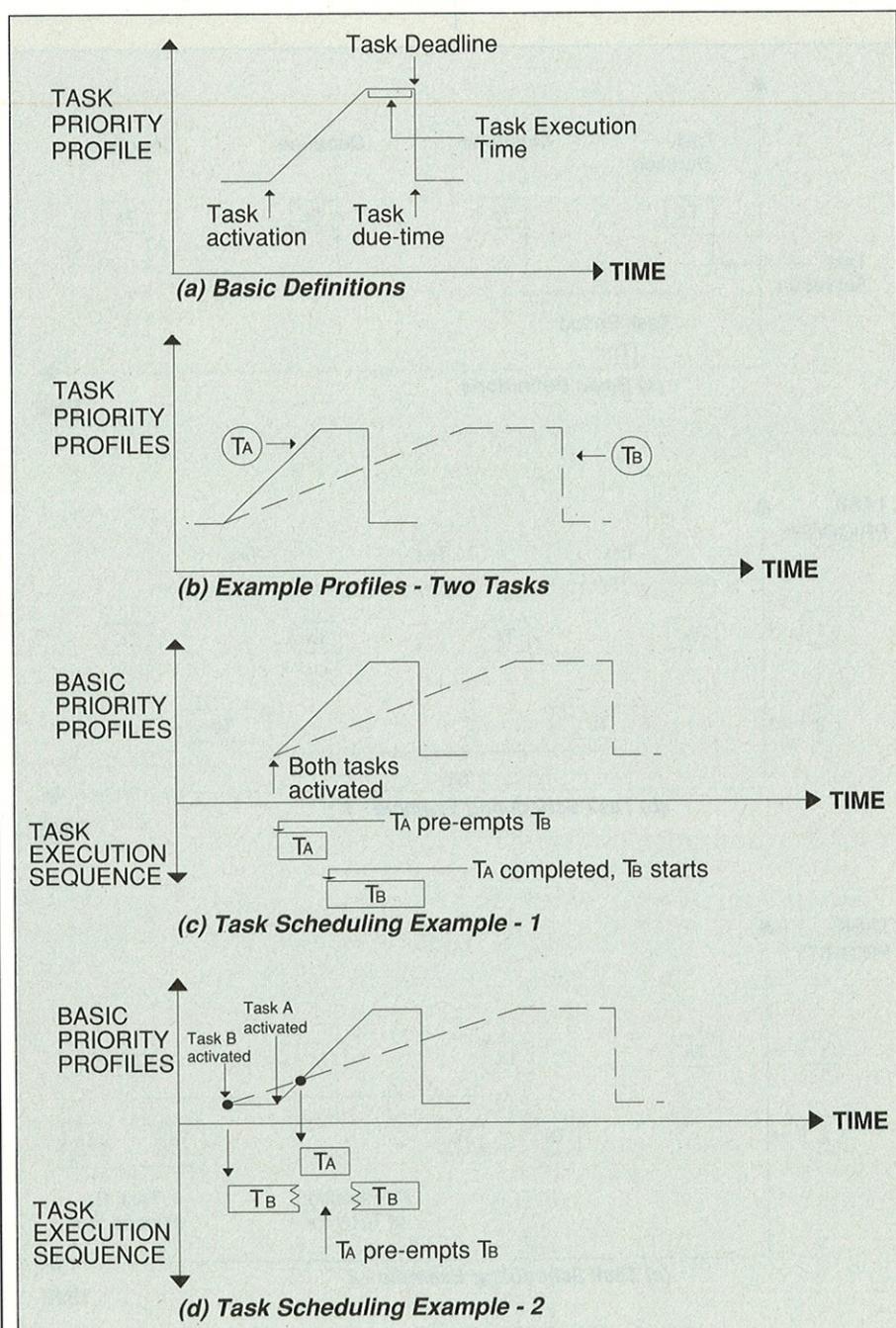


Figure 4 - Deadline Scheduling in Action



REFERENCES

1. JE Jordan, *Experiences structuring software in a periodic real time environment*, Software Practice and Experience, Vol 20(7), pp 707-718.
2. CL Liu and JW Leyland (1973), *Scheduling Algorithms for Multi-programming in a Hard Real-Time Environment*, Journal of the ACM, 20(1):46:61, 1973, pp 174-189.
3. P Molin (1990), *Implementation of an Ada run-time system supporting hard deadline scheduling*, DDC International A/S, Gl. Lundtoftevej 1B, DK-2800 Lyngby, Denmark.
4. H Piche (1988), *PORTOS - Nutzerhandbuch der PORTOS-algorithmen - Antwortzeitgesteuerte Prozessorzuteilung* (= 'PORTOS - User's handbook of PORTOS Algorithms - Response time controlled by coprocessor'), Report GPP/442/88, GPP mbH, Oberhaching.

these need to be backed up by other software tools: comprehensive development environments, debuggers for real time concurrent programs and performance analysis tools.

Research work seems far removed from the real world - recent publications induce great gloom

More specifically, we need facilities to let us build multi-tasking software from the problem point of view. The approach should be based on graphics. You should be able to describe systems as sets of concurrent tasks together with their communication and interaction features. Timing, criticality, overloading and error-handling

need to be spelt out. Integrated host/target development systems need to be produced. These should support design simulation, debugging and performance prediction using host features. For software development in the target system, a multi-tasking debugger is essential, augmented by performance monitoring and analysis tools.

This may seem like a giant wish list. But I believe that, over the next five years, improvements here will be more important than advances in executive design.

EXE

Dr Jim Cooling is a senior lecturer in the Department of Electronic and Electrical Engineering, Loughborough University of Technology. He has been deeply involved in the design, development and production of real time systems for many years, and has published regularly on the subject.

Acknowledgement: The idea of defining new categories of real time systems was inspired by comments from Nick Whitehead and Niall Cooling of Ready Systems (UK) Ltd.

PCX
Effects & Text

ADD THE POWER OF GRAPHICS TO YOUR PROGRAMS WITH HIGHLAND

DYNAMITE DEMOS with GREAT GRAPHICS **Prototype & Demo System**

Now you can demo your applications with Highland's PictureBox in almost any graphics mode including SuperVGA 800x600x256 or any of the standard VGA modes.

Use the integrated development system and the powerful Zsoft PC PaintBrush IV Plus included to create, organize, edit, and debug your demo.

Use special effects like Wipes, Explodes, Fades, Splits, Diagonals, etc. on the PCX bitmaps or on the text screens.

All commands can be controlled through mouse driven 3D button menus for intuitive and intelligent program design.

No Royalties + More Provided is a free runtime version you can distribute with your professional demo.

Supports 30+ graphic modes plus text, with special effects; cut, move, size & scale images; text on graphics; simple to use - no programming; 250+ page manual with tutorial; build fully interactive demos that respond to keystrokes, data entry, timeouts and mouse events, and much more.

PictureBox Only!

149.95
Order Today!

IMAGE Solutions!

VBase Colour Database 249.95

Turn your dBase, Clipper, C, Basic, Pascal, application into a powerful 256 - COLOUR DATABASE with little programming skills. Includes full printer support! Most any application can be converted in just a few hours.

PCX Programmers Toolkit 165.00

Includes over 75 routines to display, save, scale, and print PCX bitmapped graphics from almost any programme. Full use of virtual buffers and EMS 4.0.

PCX Effects 85.00

Add spectacular special effects (F/X) to your program. Wipe, split, crush, slide, sand, drip, diagonal, spiral, random, or explode your graphics. Requires PCX Toolkit.

PCX Text 149.00

Display blazing bitmapped text in any graphics mode as simple as text mode. You can display your text information on top of graphics.

GX Graphics 179.95

A complete graphics library supporting all graphics primitives. Use the GX Graphics Toolkit instead of the BGI or MS libraries and make your program faster, smaller and more portable across compilers - while supporting more video modes.

Pinna Electronics Ltd., APL Centre, Stevenston, Ayrshire KA20 3LR
Tel: (0294) 605296 or Fax: (0294) 68286

WE GUARANTEE Y WIDER RANGE OF SO

Over 10,000 Software Products

● COMMUNICATIONS ● DATABASES ● dBASE ADD-ON'S ● DEBUGGERS ● DEMO'S & PROTOTYPING ● DESKTOP PUBLISHING ● DIAGNOSTICS ● DOCUMENTORS ● E-MAIL ● EDITORS ● EDUCATION

Category	Product Name	Version	Description	Version	Description	Version	Description	Version	Description	Version	Description
Base	145.00	Close-Up/Customer	139.00	dSalvage Professional	149.00	Greenleaf Communications	239.00	LotusWorks	179.00	Mac Utilities '90	
386MAX Pro	79.00	Close-Up/Support	169.00	Dungeon Master	24.99	Greenleaf Superfunctions	179.00	Manifest	299.00	Macro	
ABC Flowcharter/Windows	199.00	CO/Session	159.00	Duplicator Toolkit	69.00	Gray F/X	299.00	Map Assist	329.00	MathCAD	
Above Disk Plus	65.00	CodeBase 4	195.00	Dynacom/Windows	279.00	HALO '88	329.00	MathCAD Add-On's	279.00	Mathematica/386 w/387	
Advanced Revelation	595.00	Coherent/PC	125.00	EasyCASE Plus	199.00	Harvard Graphics	195.00	Mavis Beacon Typing	495.00	Mavis Beacon Typing	
Allways 1-2-3/Symphony	99.00	Colorix VGA Paint	125.00	Easflow	149.00	Harvard Project Manager	59.00	Media Master Plus	59.00	Media Master Plus	
Alpha Four	349.00	Command Plus	79.00	Editx	129.00	hDC FileApps	75.00	Memory Mate	59.00	Menugen	
Ami Pro/Windows	325.00	Control Room	79.00	EMACS/DOS	239.00	hDC Windows Express	99.00	MenuWorks	59.00	MetaWindow Plus	
AmiDIAG	75.00	Copy II PC	27.00	EMACS/UNIX	349.00	Headroom	99.00	Mirror III	59.00	MicroTSP	
Animator	269.00	Copywrite w/Zerodisk	53.00	EnerGraphic	249.00	Hi-Screen/XL	169.00	Milestones, Etc/Windows	99.00	MKS Toolkit	
Automate/Anywhere	129.00	Corel Draw/Windows	279.00	Essential Graphics	239.00	Hijack	89.00	Monarch	229.00	MS BASIC Compiler	
Applause II	279.00	Corporate Ladder	69.00	EXP	269.00	Hooray for Henrietta	19.95	MS C Compiler	99.00	MS COBOL Compiler	
Arkanoid II	29.00	Crosstalk Mark IV	109.00	Express Publisher	129.00	Hot Wire	99.00	MS Excel/Windows	159.00	MS Pascal Compiler	
Arts & Letters/Windows	395.00	Crosstalk/Windows	97.00	Extend-a-Name Plus	73.00	Hotshot Graphics	169.00	MS Word	295.00	MS WordPerfect/Windows	
AskSam	185.00	Cruise Control	43.00	Extra-K/Symphony	69.00	HyperAccess/5	99.00	Move 'Em	169.00	MSWord + Paint	
Asyst Complete	1995.00	Dan Bricklin's Demo II	169.00	FaceLift/Windows	79.00	Hyperpad	99.00	MS Macro Assembler	299.00	MS Multiplan	
ATI/Training	59.00	DASM	169.00	Family Tree Maker	49.00	IBM Current/Windows	295.00	MS Mouse + Paint	295.00	MS Quick C	
Auto-Might	59.00	Data Junction Advanced	229.00	Fantavision	57.00	IBM Displaywrite IV/V	299.00	MS QuickBasic	59.00	MS QuickPascal	
AutoImport	129.00	Database	429.00	Fastback Plus	99.00	IBM DOS 3.x	89.00	MS Windows SDK	99.00	MS Word	
Automenu	45.00	Dataease LAN Pack	569.00	Fastlock	47.00	IBM DOS 4.x	99.00	MSWord + Paint	149.00	MSWord + Paint	
Autoroute Plus	215.00	DataPerfect	289.00	Fastlynx	85.00	IBM Fixed Disk Organizer	47.00	MSWord + Paint	149.00	MSWord + Paint	
AutoSketch v3	79.00	dBASE III Plus	327.00	FastText Search/Clipper	79.00	IBM OS2 Standard Edition	239.00	MSWord + Paint	229.00	MSWord + Paint	
Avery LabelPro/Lasers	89.00	dBASE IV/Standard	369.00	FastTrax	45.00	IBM Storyboard Plus	229.00	MSWord + Paint	229.00	MSWord + Paint	
B-Tree Filer	69.00	dBFast/Windows	329.00	File Rescue Plus	37.00	IBM Writing Assistant	129.00	MSWord + Paint	229.00	MSWord + Paint	
Baler	310.00	dBrief	79.00	Flowcharting 3	149.00	Illustrator/Windows	289.00	MSWord + Paint	289.00	MSWord + Paint	
Batcom	45.00	dBXL Diamond w/R&R	149.00	Folio Views	379.00	Impress	82.00	MSWord + Paint	82.00	MSWord + Paint	
Battery Watch	32.00	Deluxe Paint/Enhanced	79.00	Fontasy 3 Deluxe	79.00	Individual/Training	59.00	MSWord + Paint	59.00	MSWord + Paint	
BeckerTools/Windows	69.00	Derive	125.00	Fontrix	129.00	Info-XL	149.00	MSWord + Paint	149.00	MSWord + Paint	
Bitsream Fonts (each)	135.00	DesignCAD/2D	159.00	Fontspac	69.00	Info Select	85.00	MSWord + Paint	85.00	MSWord + Paint	
Blast/PC	159.00	DesignCAD/3D	229.00	Formtool Gold	63.00	Instant Recall	95.00	MSWord + Paint	95.00	MSWord + Paint	
Blinker	199.00	Designer/Windows	399.00	Formwork/Windows	249.00	Instaplan 5000	449.00	MSWord + Paint	449.00	MSWord + Paint	
Blue MAX	89.00	Desk-Link	95.00	Formwrx w/Fill & File	99.00	Intelligent Backup	95.00	MSWord + Paint	95.00	MSWord + Paint	
Brainmaker	175.00	DESKPress	229.00	FoxGraph	225.00	Kedit	109.00	MSWord + Paint	109.00	MSWord + Paint	
Brief Editor/DOS or OS/2	179.00	DESQview	79.00	FoxPro	429.00	Kermit for DOS	39.00	MSWord + Paint	39.00	MSWord + Paint	
Brooklyn Bridge	79.00	DESQview/386	99.00	FoxPro/LAN	599.00	KnowledgePRO/Windows	529.00	MSWord + Paint	529.00	MSWord + Paint	
Btrieve for DOS	389.00	dGE Graphics	229.00	Fractools	49.00	Labels Unlimited	42.00	MSWord + Paint	42.00	MSWord + Paint	
Builder	99.00	Direct Access	54.00	Framework III	339.00	Language Master	79.00	MSWord + Paint	79.00	MSWord + Paint	
C-DOC	129.00	Direct/Net	149.00	Freedom of Press	299.00	LANSpool	249.00	MSWord + Paint	249.00	MSWord + Paint	
C Async Manager	149.00	Disk Explorer	59.00	Fresh Utilities	149.00	Lantastic Starter Kit	379.00	MSWord + Paint	379.00	MSWord + Paint	
C Tools Plus	109.00	Disk Manager	69.00	Fun School III	21.00	Laplink III	89.00	MSWord + Paint	89.00	MSWord + Paint	
Cache86	39.00	Disk Manager-N	159.00	Funky Library	149.00	Laser-Ready	85.00	MSWord + Paint	85.00	MSWord + Paint	
Calendar Creator Plus	49.00	Disk Technician Advanced	110.00	GB/Executive	169.00	Lattice C	170.00	MSWord + Paint	170.00	MSWord + Paint	
Canvas/Mac	179.00	Disney Educational	21.00	GEM Artline	389.00	Le Menu	53.00	MSWord + Paint	53.00	MSWord + Paint	
Captive	30.00	Documentor	179.00	Generic 3D Drafting	329.00	LetterPerfect	139.00	MSWord + Paint	139.00	MSWord + Paint	
Carbon Copy+	99.00	DoDot/Windows	129.00	Generic CADD Level 3	329.00	LimSim	79.00	MSWord + Paint	79.00	MSWord + Paint	
Carbon Copy+ *Host	125.00	DOS Partner	69.00	Genifer III	199.00	LocoScript/PC	125.00	Multimatic Advantage II	259.00	Multimatic Advantage II	
Carbon Copy+ AutoPilot	75.00	DosUtils	65.00	Glyphix/WordPerfect	89.00	Lotus 1-2-3 V2.2	259.00	Multiscope/Windows	259.00	Multiscope/Windows	
Cardbox Plus	225.00	DoubleDisk	79.00	Gofex	59.00	Lotus 1-2-3 v3.1	339.00	Netlib Library	550.00	Netlib Library	
Certus/PC	179.00	DR DOS v5	110.00	GoScript Plus	235.00	Lotus 1-2-3/G	550.00	NetUtil	550.00	NetUtil	
Charisma/Windows	299.00	Dr. Solomon's Toolkit	49.00	Grammatik IV (UK)	98.00	Lotus 1-2-3 for UNIX/386	525.00	Netware C Interface	525.00	Netware C Interface	
CheckIt	85.00	Dr. Switch	79.00	Grammatik IV/Windows	69.00	Lotus Agenda	278.00	Neuroshell	278.00	Neuroshell	
Clarion Pro Developer	549.00	Drafix CAD/Windows	495.00	Grapher	149.00	Lotus Freelance Plus	279.00	New Print Shop	279.00	New Print Shop	
Clear Plus/C or dBASE	129.00	DrawPerfect	329.00	Graphics Link Plus	139.00	Lotus Magellan	99.00	NewMaster II	99.00	NewMaster II	
Clipper v5	325.00	DS Backup Plus	72.00	Grasp	109.00	Lotus Symphony	419.00	Norton Anti-Virus	419.00	Norton Anti-Virus	

THIS IS A SMALL SAMPLE OF OUR RANGE

WE'LL BEAT ANY NATIONAL

Software Paradise is an independent software reseller representing more than 2000 manufacturers with over 10,000 software products for IBM Personal Computers (and the Apple Macintosh) running DOS, OS/2 and UNIX. We offer the most comprehensive range of PC software in the world and have been supplying both users and programmers with quality products and the service at the lowest prices since 1986. "The Discerning Person's Guide to Software" is the most complete Buyers's Guide to PC software available and is published quarterly by Software Paradise. Call us on (+44) (0222) 887521 or complete the coupon to receive your **FREE** copy.

FREEPHONE 0800 378·873

ORDER LINE ONLY

**PLACE YOUR ORDERS FREE OF CHARGE
FROM ANYWHERE IN THE UK**

ALL OTHER ENQUIRIES TEL: (+44) (0222) 887521 (10 LINES) FAX: (+44) (0222) 862209 (24 HRS)

YOU WON'T FIND A SOFTWARE ANYWHERE*

for DOS, OS/2, Mac & UNIX

IAL • EXPERT SYSTEMS & A.I. • GRAPHICS • HARD DISK BACKUP • INTEGRATED SOFTWARE • LANGUAGES • LINKERS • MAC SOFTWARE • MATHS & STATS • MEMORY MANAGEMENT

129.00	Norton Backup	69.00	Pizazz Plus	79.00	RTLINK Plus	329.00	Toolbook/Windows	257.00
125.00	Norton Commander	79.00	PKZIP/PKUNZIP	54.00	Saber Menu/DOS	89.00	Top Priority	79.00
39.00	Norton Editor	49.00	PlanPerfect	235.00	Saber Menu/Windows	449.00	Touch & Go	20.00
229.00	Norton Guides C, OS2 etc	57.00	Playroom	37.00	SCO XENIX/386 Complete	1275.00	Tree Diagrammer	65.00
345.00	Norton Utilities v3	109.00	Pink's 36 Plus	349.00	Security Guardian	229.00	Turbo Asynch Plus	139.00
75.00	NoSquint	45.00	PolyBoost II	65.00	Seemore 1-2-3	65.00	Turbo C++	125.00
595.00	Nutshell Plus	189.00	PolyMake	110.00	Service Diagnostics Kit	395.00	Turbo C++ Professional	205.00
35.00	ObjectVision	299.00	PolyShell	79.00	Show Partner F/X	245.00	Turbo Pascal	79.00
55.00	Object Professional	99.00	Popdrop Plus	75.00	ShowText	149.00	Turbo Pascal Professional	149.00
45.00	OmniPage-386/Windows	529.00	Portex Professional	117.00	Sidekick Plus	97.00	Turbo Professional	79.00
45.00	Open Access III	479.00	Power BASIC	79.00	Sidetalk	79.00	TurboCAD	129.00
35.00	Opt-Tech SORT	125.00	PowerMenu	69.00	Sideways	39.00	Twist & Shout	49.00
185.00	Option Board Deluxe	105.00	Powerpak/Windows	69.00	Sigma Plot	445.00	Type Manager/Windows	79.00
499.00	Optune	59.00	PreScript Deluxe	295.00	Silvercomm Library	189.00	Typing Tutor IV Plus	39.00
129.00	ORCAD/PCB	1395.00	Presentation Task Force	249.00	Sim City	26.08	Typografica Prime	49.00
160.00	ORCAD/SDT	495.00	Primeline Personal	75.00	SitBack	62.00	UI Programmer v2	279.00
169.00	Org Plus Advanced	85.00	PrintAPlot	165.00	SiteLock w/Anti-Virus	329.00	UI Touch & Go	235.00
299.00	Overlay(s) for Clipper	99.00	Printcache	95.00	Smalltalk V/286	129.00	UltraScript	125.00
57.00	P.D.Quesus for 1-2-3	59.00	Printer Assist	149.00	Smalltalk V/PM	295.00	UltraScript PC Plus	245.00
239.00	PageGarden	63.00	Printmaster Plus	49.00	Smarternet 240	179.00	Ultradivision	79.00
229.00	Pagemaker/Mac	569.00	PrintQ	89.00	Smart System	469.00	Vcache	53.00
469.00	Pagemaker/Windows	453.00	Printrite	85.00	Smartware II	469.00	Ventura GOLD/Windows	549.00
259.00	Paradox v3.5	379.00	Procomm Plus	72.00	Soft Kicker Plus	95.00	Vermont Views	349.00
39.00	Paradox Engine	269.00	Procomm Plus w/Viewdata	107.00	Soft Type/Windows	129.00	Virex-PC or Mac	79.00
19.00	Paradox SQL Link	269.00	Professional File	169.00	Softbytes	59.00	Virus Hunter	99.00
229.00	Pathminder Plus	79.00	Professional Write	229.00	Software Bridge	85.00	VirusSafe	79.00
85.00	PC Anywhere III/IV	110.00	Professor DOS	57.00	Software Carousel	57.00	VM/386	145.00
92.00	PC Guide/Windows	429.00	Project Calc/1-2-3	139.00	Sourcer/486 w/BIOS	125.00	VolksWriter 4	125.00
109.00	PC Paintbrush IV Plus	159.00	Project Scheduler/Windows	449.00	SourcePrint	69.00	Vopt	45.00
159.00	PC Tools Deluxe	79.00	Project Workbench	795.00	SPF/PC	169.00	Watchdog	185.00
298.00	PC Write	85.00	Prokey Plus	79.00	Spirnrite II	89.00	Where in Europe	39.00
349.00	PC-Browse	55.00	PVCS Professional	449.00	Sprint	135.00	Who-What-When	159.00
59.00	PC-File 5	79.00	PVCS/LAN	1595.00	SPSS/PC+	99.00	Windows Workstation	595.00
57.00	PC-Fullbak Plus	69.00	Q&A	239.00	StatGraphics	429.00	Wings/Windows	329.00
59.00	PC-Globe	69.00	QDOS-II	49.00	Sterling Accountant Plus	329.00	Wonder Plus	59.00
275.00	PC-Guard/D Plus	137.00	QEdit TSR	79.00	Sterling Financial Controller	429.00	Word for Word Pro	99.00
75.00	PC-Kwik Power Disk	59.00	QEMM/386 w/Manifest	57.00	Stonybrook Modula-2	229.00	WordPerfect	269.00
253.00	PC-Kwik Power Pack	83.00	Qram w/Manifest	57.00	Storyboard Plus	229.00	WordPerfect Addn'l Users	149.00
299.00	PC-Lint	89.00	Quattro Pro	269.00	Super-PC-Kwik	53.00	WordPerfect Office	339.00
99.00	PDC PROLOG for DOS	199.00	Quick Schedule Plus	59.00	Superbase 4/Windows	449.00	WordPerfect/UNIX	329.00
79.00	Pegasus SU Modules	205.00	QuickPak Professional	129.00	Supercalc 5	75.00	WordScan Plus/Windows	795.00
89.00	Pegasus Senior Modules	429.00	R&R Report Writer	135.00	SuperFonts	125.00	Wordstar 2000+	249.00
139.00	PerFORM Pro/Windows	269.00	R&R Clipper/Fox Module	49.00	Superkey	55.00	Wordstar Pro	235.00
325.00	PerFORM/GEM	159.00	R&R Code Generator	99.00	SuperPrint/Windows	149.00	XENON II MagaBlast	21.70
299.00	Personal COBOL	129.00	R:Base for DOS	379.00	SuperProject Expert	459.00	XtreeNET/VELS	179.00
289.00	Personal Measure	69.00	Rapidfile	195.00	Surfer	339.00	XtreeNET/Advanced	299.00
529.00	Personal REXX	129.00	Reader Rabbit	35.00	Switch-It	72.00	Xtree Pro	63.00
149.00	Perspective Junior	129.00	ReadRight/Windows	429.00	Systat w/Graphics	599.00	Xtree Pro Gold	83.00
79.00	Perfmaster Advance	1639.00	Reflection 2+	199.00	System Sleuth	85.00	XYWrite III Plus	269.00
185.00	PFB:First Choice	123.00	Reflection 4+	229.00	System WinSleuth/Windows	85.00	XYWrite III Plus/LAN	445.00
133.00	PFS:First Graphics	117.00	Reflex v2	179.00	Take II Manager	79.00	ZINC Library	179.00
45.00	PFS:First Publisher	125.00	Relay Gold	179.00	Talking Windows	235.00	Zortech C++ Video Course	269.00
59.00	Picture Publisher	449.00	Remote 2 Complete	129.00	Timeline v4	479.00	Zortech C++ Developer	269.00
129.00	Pixie/Windows	169.00	RM/FORTRAN	399.00	Timeworks	85.00		

MISCELLANEOUS • NETWORK SOFTWARE • OCR & SCANNING • OPERATING SYSTEMS • PIM'S & HYPertext • PRINT SOFTWARE

CALL US FOR ANY ITEMS NOT LISTED

LY ADVERTISED PRICE+

OUR POLICY

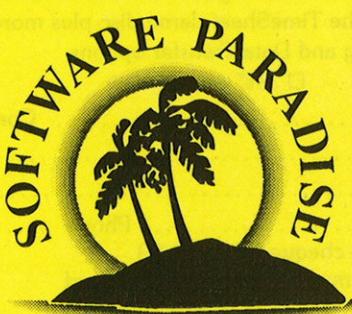
- All prices and manufacturer's specifications are subject to change without notice.
- All prices exclude carriage and VAT.
- All orders subject to availability.
- Please specify media size when ordering.
- Goods are not offered on a trial basis.
- Goods are offered for sale subject to our standard conditions of sale which are available on request. E&OE.



- Provide written proof of any U.K. mail-order software company who supplies a wider range of IBM PC software than we offer in our catalogue within 14 days of this advertisement and get yourself a FREE copy of PC Tools Deluxe!

+ We are always delighted to be given the opportunity to beat other nationally advertised prices (at the time of placing an order) subject to the same terms and conditions of the company being beaten.

All Trademarks and Registered Trademarks are hereby acknowledged.



The Software Experts

Software Paradise, Avenue House, King Edward Avenue, Caerphilly, Mid Glamorgan, CF8 1HF, UK

**FREEPOST
— TODAY —
NO STAMP NEEDED**



Please send me my *FREE* 32 page copy of
'The Discerning Person's Guide to Software'

NAME

COMPANY

ADDRESS

POSTCODE

TEI

Post now to: Software Paradise, FREEPOST, Avenue House
King Edward Avenue, Caerphilly, CF8 1ZZ

EYE 291

Time\$heet™

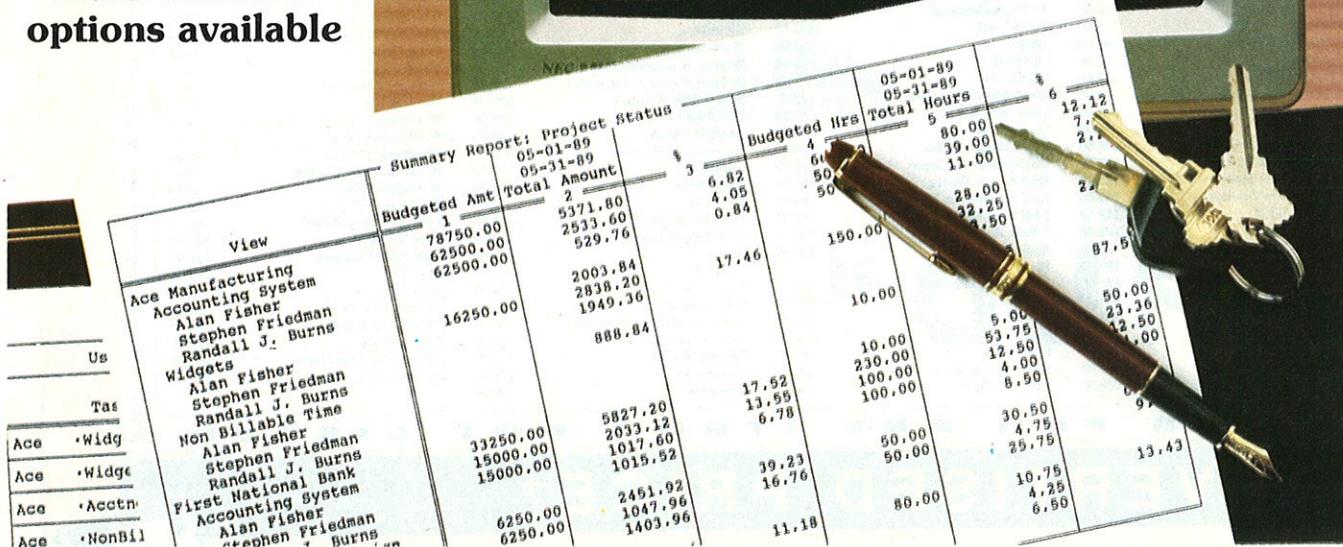
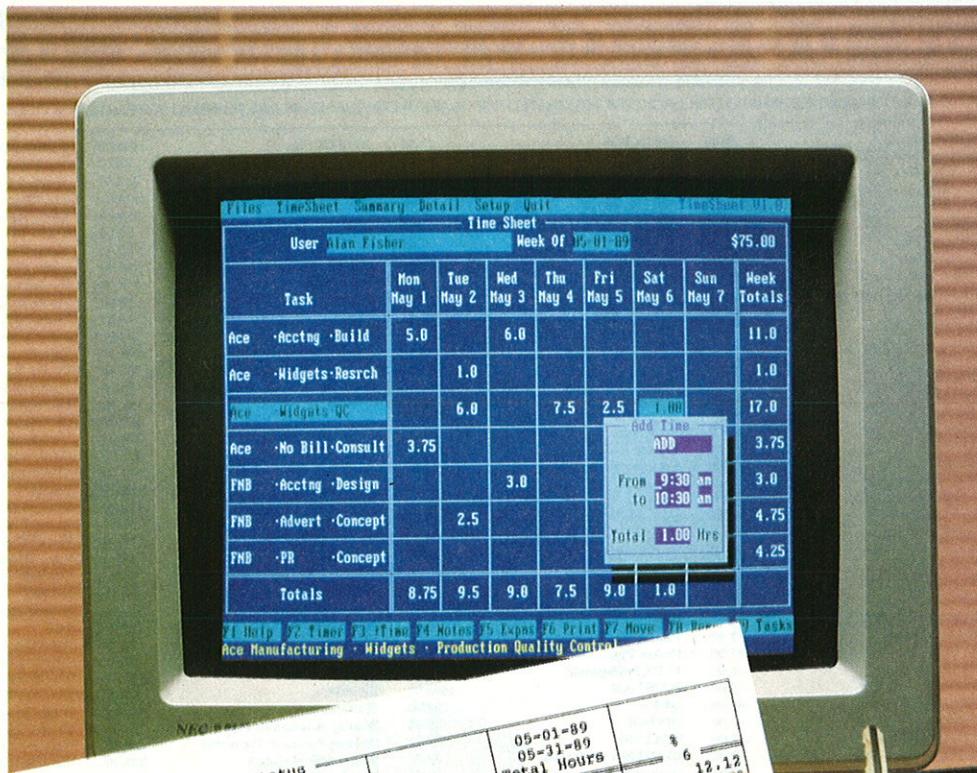
Just £99!

P R O F E S S I O N A L

**Intuitive and
Flexible —**

**Time and
Expense
Tracking and
Reporting
for Projects
and Clients**

**★ Networking and
Data Transfer
options available**



**DEEPAK
SAREEN
ASSOCIATES**

**FREEPOST,
28 Preston Road,
Wembley HA9 8BR.
Tel. 081 908 2423
Fax 081 904 6668**

- Rush me TimeSheet Professional right away at only £99.
Please add £5 handling and VAT. total £119.60
- Send me the TimeSheet demo disc plus more information on TimeSheet and the Networking and Data Transfer options.
- 5½" discs 3½" discs

Name Company

Address

Postcode Phone

.EXE 2/91

- I enclose a cheque for £119.60
- Please bill my Visa/Access/Amex Card
-

Expiry date

Signature

**SEND NOW FOR
FREE DEMO DISC**

CIRCLE NO. 498

A Video labelled 'X'

With the price of 'live' training courses, the training video offers an attractive alternative. Willie Watts has been couch potato-ing through an X windows course.

The All-Hands-On X Video Workshop is a new US product comprising 11 videos (each about 20-30 minutes long) in two volumes: *An Introduction to the X Window system* and *Programming with XLIB*. Accompanying documentation consists of a copy of the standard three volume *X Window System User's Guide*, and a workbook consisting of a transcript of the videos with exercises inserted at relevant points. This Christmas, while the rest of the world was snivelling over *ET*, I slogged through the adventures of Bif and Marilyn.

Bif Boston (*sic*) and Marilyn Megabucks (*sic again*) are our hosts for the course. Bif, his slightly dopey face jaundiced by the conversion to the British PAL video standard, is the Comic Relief. Handsome Marilyn, whose Sino-American complexion stands up rather better to this treatment, plays the Love Interest. Sat together behind a desk, as though reading News At Ten, they indulge in a scripted repartee which finally proves, if ever there was any doubt, that there is a gap of about 15 degrees between the British and US sense of humour. (**Marilyn:**

'Hey, Bif, what are you so happy about anyway?' **Bif:** 'I finished my beautiful thing.' **Marilyn:** 'I beg your pardon?' **Bif:** 'My beautiful picture! Take a look.' **Marilyn:** 'A house! What a nice house! It makes the Mona Lisa look like a kid's drawing.' *She looks to camera conspiratorially and rolls her eyes.* 'Or is it the other way around?'

The three videos of Volume 1, the introductory section, not only introduce us to Bif and Marilyn, but contain a mass of in and extra-studio comic business. The concept of the graphics server, for example, is illustrated by an acted out piece set in a burger bar, where the food-orders represent key-strokes, the food is the graphics and there are three kitchens (local, LAN and WAN) illustrating the effects of varying response time. In places the surrealism is cranked up to overload. The X philosophy is explained by, among others, a vacuum cleaner salesman, a Terry Jones style transvestite and a naked man standing in a shower cubicle.

None of the material in Volume 1 is very technical, presumably because it is intended to be palatable to Suits. However, you end up with a very complete picture of what the X components are, and how they all fit together. If you are hazy on the relationship of window manager products like Motif to X itself, then this will sort you.

In Volume 2 we get down to code level. Bif and Marilyn come into their own here, maintaining a constant sexual tension, slightly reminiscent of Steed and Mrs Peel in *The Avengers*. Whenever the script begins to drag ('For drawing figures that are made of lines, as opposed to filled figures, we have XDrawLine, to draw a single line, XDrawLines, to draw multiple connected lines, and XDrawSegments, to draw multiple non-connected lines. We also have XDrawRectangle...') Bif will liven proceedings, perhaps by climbing into the Xwindow he has just coded - a CSO

effect - in order to impress Marilyn (no chance: she just does some more eye-rolling). There are also plenty of animated diagrams, a token 'real programmer' called Warren Watcher (but we can tell that he's really an actor; he changes his T-shirt too often) and a truly dire cartoon character called Mr Event Switch... but B and M are the real stars.

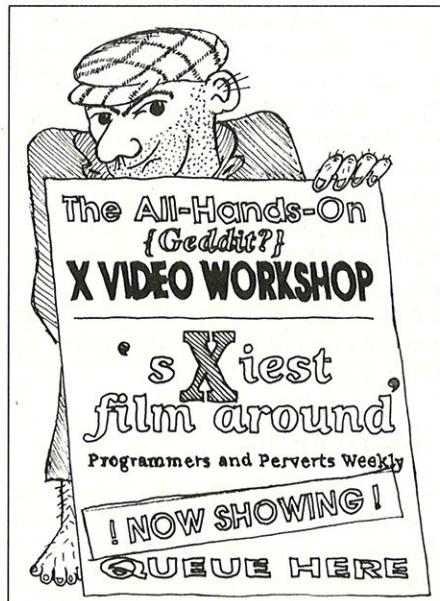
The second volume, then, covers all the important XLIB C calls, explaining their function in the context of example programs. It builds up from elementary topics, such as program structure, through to font manipulation, colours (the module is filmed in black and white) and bit images. Note that the material deals with the lowest X API level - no coverage of 'widgets' or 'toolkits' here.

Since I did not have the necessary computer equipment to hand, I missed out on one major component of this course: the 'Hands-on' practicals. But that the videos sustained my interest, more or less, *without* getting to try it all out.

To summarise: although the (intentional) laughs are not up to the standard of an ex-Python or a Smith & Jones production, I enjoyed watching the *X Video Workshop* a lot. And when I'd finished wincing at gauche Bif and squirming at Marilyn's coquettishness, I discovered that I knew rather a lot about X. Product performs to spec. **Marilyn and Bif:** (*waving*) 'Bye for now!'

EXE

X Video Workshop is distributed by MMT UNIX Education (071 278 6211). Prices are as follows: Volume 1 (three videos) £1000, Volume 2 (eight videos) £1700. Documentation sets are priced at £65 and £125 respectively, and the videos are available in both VHS and Beta.



Memoirs of a Frontier Programmer

What's over 30 years old, has a CHAPTER 0 and goes hoot? Ask George Ruscoe, a veritable software old-timer, who has been looking back to his first experiences of a high-level language.

Recently I was giving training in some EDI software tools to a group of young Graduate Programmers. It came as something of a shock to realise that none of them was born when I first learned to program using Mercury Autocode. Perhaps it was time to begin my memoirs! Here is a start.

Mercury Autocode was first implemented on the Ferranti Mercury in March 1958 by a team lead by Prof R A Brooker, who had also developed the previous Autocode for the Mark I computer. (He subsequently went on to do work on phrase structure and formal language description, and this led to the Compiler Compiler, which was the first successful attempt at a formal language for the transparent description of syntax and semantics.) The program in Figure 1 shows a solution of the familiar Triangle problem. If you have read a previous Third Side article on the original Manchester Autocode (.EXE, April 1990), you will notice many similarities, but also some significant differences.

Arithmetic

The notation of Mercury Autocode owed a good deal to the conventions of mathematics, so that variables were represented by

letters, such as X and Y, and subscripted variables by A1 or AK. This was a leap forward from the original notation, which only allowed variables called V1 to V8192 - effectively one big array. Arrays were declared by the directive A->10 which set

Several of the ladies were of the opinion that high-level languages would never catch on

aside 11 elements named A0 to A10. In addition there was a set of *primed* variables, A', B' etc, and a special variable π (Pi) initialised to 3.14159265 - the latter reflecting the fact that most programs written for the Mercury machine were mathematical and engineering calculations.

These *main variables* (arrays) and *special variables* (the single letters) were represented by 40-bit words and processed in floating point hardware. There were also 12 *indices*, held in of 10-bits, and represented by the letters I to T, so that only the remaining 14 letters could be used for 'variables'. The use of brackets was very restricted, and there was normally no symbol for multiplication, so by convention A2 and AK both referred to array A with a subscript, while 2A or KA implied multiplication. The most complicated term that could be written was 3JKXA (N+5) / Y, which in modern terms is $3 \cdot J \cdot K \cdot X \cdot A [N+5] / Y$. There was often a need to form preliminary sums, especially for a denominator. It was a convention to use certain variables (W1, W2 etc) only for the working intermediate variables; I have used W in this way in the example program.

The only use of brackets was for a *complex subscript*, which took the form *index + integer*. An integer 'arithmetical instruction' could only include integers (see Figure 2), but if the left hand side was a real variable, the right hand side could contain either type - an early form of mixed mode expression. The language also provided some standard functions, the common ones being named and preceded by the Greek letter phi, eg $\phi\text{SQRT}(X)$. This was an improvement on the F1 notation of the original Autocode, although many of the less common functions were still given numbers; for example $\phi 16$ would transpose a matrix and $\phi 8$ would print it.

Input/Output

In general, the old system of abbreviating statement words to one letter had been abandoned. J had expanded to JUMP; H (halt) was now END and input was by

<pre> TITLE TRIANGLE CHECK - DATA = THREE SIDES CHAPTER 0 READ (A) READ (B) READ (C) W=B+C JUMP 1, A≥W W=C+A JUMP 1, B≥W W=A+B JUMP 2, W>C 1) CAPTION THIS IS NOT A TRIANGLE </pre>	<pre> END 2) JUMP 3, A≠B JUMP 3, A≠C CAPTION THIS IS AN EQUILATERAL TRIANGLE JUMP 9 3) JUMP 4, A=B JUMP 4, B=C JUMP 4, C=A CAPTION THIS IS A SCALENE TRIANGLE JUMP 9 4) CAPTION THIS IS AN ISOSCELES TRIANGLE 9) END CLOSE </pre>
--	---

Figure 1 - Solution to the Triangle problem



A->99	allow A0 to A99 as an array
I=2MN+M+N+1	integer
X=2NA1-MA2+5MNA (N+2) /D	float
Y=2π φSIN (X)	standard function
(U, V)=(X, Y) * (A, B)	complex arithmetic
Φ6(X) W, N	transfer N variables from Auxiliary store at X to working store at W
A'=Φ11(B', C', U)	matrix A=B+C of size U, were A', B' & C' are locations of matrices in auxiliary store

Figure 2 - Sample Arithmetic Statements

READ. Output had also been much improved with the PRINT command (see Figure 3) which would take any variable expression X and convert it for output with M characters before the point and N after it, suitably rounded. Numbers would always be followed by two spaces, but additional control could be gained by SPACE and NEWLINE instructions. On input, a number was terminated by two spaces or carriage return and linefeed, so that input and output were compatible. Remember that in this era, input was prepared by punching a five channel tape on a teleprinter, and the output was produced on a punched tape which was then printed out offline.

There was a facility for printing the result of any calculation by placing a question mark at the start of the statement. This was regarded as a form of diagnostic print (called a query print) and could be switched on or off by setting hand switch four on the console.

In the example program, the program tape would be read in and the program compiled. When CLOSE of CHAPTER 0 was encountered, execution would start at the first instruction of that chapter. At this point, the computer would try to read in a data tape which you should by now have loaded. Two features shown in the program were later additions. The TITLE directive was introduced so that the output could be identified! This was acted upon as the tape was read in, and the contents of the following line were copied to the output punch. CAPTION was a similar run-time instruction which caused the content of the following line to be output at that point in the program. Prior to this, your output would consist of a set of numbers only, unless you

inserted machine code instructions in your program to output a character at a time.

Organisation and Control

Mercury Autocode was conceptually divided into four areas: Arithmetic & Storage, Input/Output, Program Organisation and Control. The first two groups we have already seen. The simple Control instruc-

A table gives the time taken up by each instruction and operator - in micro minutes

tions of unconditional and conditional jumps are much the same as in the original Autocode. There were symbols for 'greater than', 'greater or equal', 'equal' and 'not equal', and you could compare two variables or a variable and constant. Note that if you wanted to test for 'less than' you had to reverse the condition and jump the other way! Labels were separated from the statements by a right bracket (rather than the colon of modern languages).

One of the major new features was the cycle (loop). This consisted of a pair of instructions: I=N (S) M and REPEAT (shown in Figure 4) which came at the start and end of the loop. The step S could be either

```
READ (I)
READ (X)
PRINT (X) M, N
SPACE
NEWLINE
HALT
HOOT n
CAPTION
LINE OF CAPTION...
```

pause with hoot, wait for key
n=1 to 8 in the major scale

positive or negative, and the items were all indices or constant integers. You had to make sure that M-N was a multiple of S, as the terminating test was for equality. Loops might be nested to a depth of eight and were one of the first high-level control instructions. Another interesting control statement was the special form of test, CHECK, which had the effect 'jump to label 3 if E is greater than the absolute value of (X-Y)'. This was used to test for convergence.

The other important control feature - the subroutine - was implemented in a number of ways. The design of the first was more or less determined by the limited size of the machine. The main store (which was made of magnetic cores) was limited to 1024 words of 40-bits each. Backing store was held on magnetic drums. Half of the main store could contain your program, the rest held data. Autocode programs were restricted to 832 instruction registers (these were medium sized 20-bit words) and the main variables were restricted to 480. You could declare A->479, but that was your lot.

So far as the program code was concerned, any large program was split into chapters, headed CHAPTER 1 and so on, each of which had its fixed place on the drum. The space not used by these was available for auxiliary variables up to (10,752 - 512N), given N chapters. As they were compiled, these chapters were stored away, but when Chapter 0 arrived (at the end), it was compiled and put into execution, starting at the first instruction. A small program might be held entirely in this chapter, but usually it was necessary to split. You transferred control to another chapter with ACROSS 1/2, which meant execute chapter 2 starting at label 1. Thus multiple entry points were provided by different labels.

To treat a chapter as a subroutine you could use DOWN 1/3 and return to the calling chapter with UP. If you wanted to look after your main memory variables during the call, you could use PRESERVE before DOWN and RESTORE immediately after. This would guarantee that the subchapter did not disrupt the variables in the calling chapter, although any values set before the DOWN were still available in the sub-chapter. To return values, you have to place them in the auxiliary variables on the drum before the RESTORE took effect.

It was very inefficient and slow to swap chapters every time a small routine was called. This problem also dogged standard library functions. Although some of these were coded inline, most were pulled in from the drum when needed. If you had room in your chapter, you could list the

Figure 3 - Sample Input/Output Statements

I=1 (1) N	start loop
PRINT (AI) 5, 2	
REPEAT	end loop
JUMP 9, I=N	conditional
CHECK (X, Y, E, 3)	test difference X-Y
END	stop
L)=3)	L becomes address of label 3 JUMP 20 to subroutine
3) ...	return label (for this call)
20) ...	
JUMP L)	the subroutine return from subroutine
L)=N)	if N=3, L is address of label 3
JUMPDOWN 50	subroutine entry
RETURN	
0)	Label to trap READ fault (32)
100)	Label to continue on other faults

Figure 4 - Sample Control Statements

functions in order of priority, and those which would fit - known as 'quickies' - were incorporated.

It was possible to include a small subroutine in a chapter, but you had to manage the call and return yourself. It was possible to store a label value in an index, say L)=3), before entering the routine with a JUMP. The return was achieved with a JUMP (L). Eventually a new instruction, JUMPDOWN, was added. This noted its own address on the stack, and was paired with RETURN, which worked as you would guess. Parameter passing, however, never progressed beyond setting global variables as detailed in the specification of the routine.

Further Features

The later versions of Mercury Autocode were called 'CHLF Autocode', named after the sites at Cern, Harwell, London and Farnborough where the various enhancements

had been developed. In the final release, CHLF3, a form of internal procedure had been invented. This was declared like this ROUTINE 50 and entered by JUMP-DOWN (R50) or JUMPDOWN (R50/3). It boasted its own set of internal labels, distinct from those of the main routine - but still shared the same variables.

Other advanced features offered by the language included random, complex and double precision numbers, matrix manipulations (all matrices were held on the drum) and step by step integration of differential equations by the INT STEP instruction. A considerable library of programs and routines was built up, mainly of a mathematical nature.

There was a lot of interest in the size and speed of programs. For instance, there is an instruction N)=M) for setting a switch label N from an integer value M, but the manual tells us that 'the execution time of n)=m) is

very much longer (18 millisecs) than that of n)=3) (120 microseconds).¹ There is also a table giving the space and time (in micro minutes) taken up by each instruction and operator! The main use of this was to emphasise that a divide was 12 times as slow as a multiply and 60 times as slow as an add, while chapter changing and input/output in general were very time consuming (as one might expect). However, it was often a matter of shoe-horning a program into the least space, and trying to optimise speed, so this information was valuable.

Female Luddites

When I first met Mercury Autocode in 1962, it was already well established, although the CHLF3 version had yet to appear. However, several of the ladies with whom I worked (they were in a majority in the department) were of the opinion that these 'high-level languages' would never catch on with professionals. They continued to code away busily in purest assembler. But FORTRAN was already available, and soon we were beginning to use ALGOL on the English Electric KDF9.

Meanwhile one of my colleagues came back from the USA with tidings of a new language. It had variables called A1, X9 and READ and PRINT statements; GOTO and IF jumps, FOR and NEXT loops and GOSUB and RETURN. At first sight it didn't seem to have much over Autocode, apart from more general arithmetic expressions. What I didn't realise at the time was that you used it directly from a time sharing terminal. It was called BASIC, and by early 1968 I found myself teaching it in half day courses every Wednesday morning.

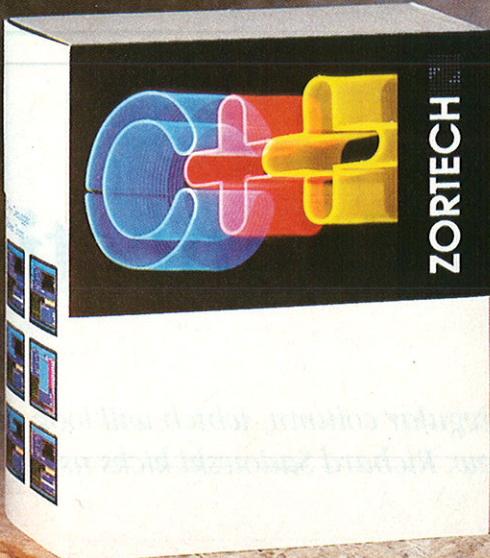
Since then, I have spent a lot of time wrestling with BASIC and FORTRAN and PL/I and C, but I still look fondly back to the days when you took your paper tape down to the machine room and fed it into the beast yourself. Of course, the time sharing system brought immediacy to the user, and the advent of the PC has put far more power on the desk than ever we had in that huge machine room. But somehow it's not quite as impressive, and anyway with modern computers without refrigeration units, there is nowhere to keep the milk cool. Ah, the price of progress!

EXE

George Ruscoe began his programming career at AEI in Manchester. In more recent times he has worked from home with a new-fangled PC, and is currently employed by software consultants Humphreys & Dean.

ACROSS 1/2	to label 1 chapter 2
CHAPTER 2	
1) ...	
PRESERVE	
DOWN 1/3	
RESTORE	
...	
CLOSE	textual end of chapter
CHAPTER 3	
1) ...	
...	
UP	
CLOSE	

Figure 5 - Sample Organisation Statements



For total portability right now, your choices are a bit limited.

ONLY ZORTECH GIVES YOU TOTAL PORTABILITY TO MS-DOS, WINDOWS, OS/2, DOS 386, UNIX 386. AND MACINTOSH. RIGHT NOW.

WHAT PLATFORM WILL YOUR APPLICATIONS BE RUNNING ON THIS TIME NEXT YEAR?

Now that C++ has made significant inroads into mainstream application development, you must choose the C++ compiler that is going to be the performance and portability leader now...and years to come. Since 1986, we've been testing, refining and perfecting our family of C++ products tirelessly. The results are what you are reading about: a fully portable family of C and C++ products available right now.

WE INTEND TO KEEP UP THIS PACE.

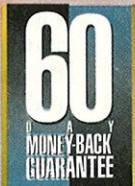
In the past few years, Zortech has quietly become an industry standard earning acclaim from Byte, Computer Language, Dr. Dobbs, and many industry experts for our solid implementation, market-leading performance, and strong, relentless innovation.

WHATEVER PLATFORM YOU CHOOSE, CHOOSE ZORTECH PERFORMANCE.

With Zortech, no matter what platform you choose to develop on, a substantial performance increase will be realized due to the quality of the original Zortech C++ implementation. For instance, on MS-DOS, your program will compile up to 35% faster with the resultant code running up to 45% faster and 25% smaller than Turbo C++. But we don't stop there. On each platform, Zortech C++ is designed to improve your productivity where it really

matters... in the development cycle.

For example, using the DOS/MS Windows package, you can edit, compile, link and debug the largest MS windows applications...and never once leave the Windows environment. All this plus complete plug-and-go to CNS C++/View and Commonview II! And, of course, these advantages are also available on the OS/2, DOS 386, UNIX, and Macintosh platforms.



CHOICES. CHOICES. CHOICES...

If there is one constant in C++ development, it's change. The Zortech family of products keeps your options exactly how they should be. Open. To find out more, call today.

ORDER HOTLINE:

44-81-316-7777

Zortech

Industry Standard. Industrial Strength.

ZORTECH, INC., 4-C GILL STREET, WOBURN, MA 01801 TEL: (617) 937-0696 FAX: (617) 937-0793
ZORTECH, LTD., 58-60 BEREDFORD STREET, LONDON SE18 6BG TEL: +44-81-316-7777 FAX: +44-81-316-4138
SUPPORTED SYSTEMS INCLUDE COMPAQ, ISA/EISA, IBM PS/2 AND 100% COMPATIBLES

* Source: MS-DOS, C++ Compilers, A comparison by the Ladd Group, (c) 1990

CIRCLE NO. 499

The Polish Connection



A new year, and a new .EXE regular column, which will look at software development issues from a foreign point of view. Richard Sadowski kicks us off with a Polish perspective.

Poland has a long and honourable tradition of computer science. The father of computer calculation was the Polish scientist Lukasiewicz, probably best known for so-called Reverse Polish Notation. Another claim to fame is that a Pole, a Mr Karpinski, invented one of the first minicomputers 20 years ago. But Karpinski's project was abandoned because of the political factors: it didn't fit into the Eastern (= Soviet) standards.

Karpinski was forced to forget his dreams and to work as a farmer for many years - but the story has a happier ending. With the new wave of the Solidarity movement he was able to return to his original interest. Now he runs his own company abroad and is a respectable engineer and manager. His story is very typical of our older computer engineers and programmers.

Before the era of PCs, the only machines used in Poland were the ODRA (a Polish version of ICL 1900), RIAD (a Russian IBM 360-alike mainframe), SM-60 and MERA (both based on DEC's PDP-11). These were situated in the academic institutions and large Government computer centres. At that time, the demand for computer science graduates was slack. There were no economic reasons to change this situation, as the jobs market was controlled by a few state-owned companies.

The first wave of the computer 'madness' arrived in Poland in 1985. Almost every technical magazine printed a short course on BASIC. Everyone wanted to become a programmer. The excitement only calmed down when people started to understand that it was impossible to write all the software they needed from scratch.

Poland was flooded with hundreds of home and personal computers. The most important machine of those times was the ZX Spectrum which, thanks to its low price and the fact that it could be exported from the West without a licence, enjoyed

tremendous popularity not only among the game enthusiasts but also in the serious scientific institutions and state companies. For people used to fighting with card punch machines, even this was a real step forward.

The shortage of software was overcome by not introducing any software copyright laws

Fortunately the Spectrum soon began to give way to more serious personal computers. Again, price and COCOM restrictions dictated choice of machine. Ninety per cent of the Polish market was cornered by cheap Taiwanese PC clones.

However, computers were not yet viable for everyday applications. There were two reasons for this. The first was cost: the price of an average PC was approximately equivalent to a programmer's wages for 10 years. The second problem was lack of software. It took about five years to reduce significantly the price of computers - they are now worth about six months' salary. The shortage-of-software obstacle was simply overcome by the old Communist government, which chose not to introduce any software copyright laws.

All kinds of software were copied freely. This situation had some advantages. Few companies and developers could afford to pay the full price for the software, as Western software houses didn't discount their products when selling in Poland. However,

there were big drawbacks as well. Polish software developers could not sell their products. Anything that reached the market was instantly pirated. We hope that this situation will change soon, when our Parliament sets a new copyright law. For the time being, most of the programs sold in Poland are protected by various software and hardware keys.

In 1990, with the easing of COCOM restrictions, a second wave of computer equipment started to come to Poland. Even though PCs continue to dominate, an ever greater variety of systems is penetrating the Polish market. There are VAXs, Unisys machines, ICLs, IBMs and others. Lack of software to run on them remains a major problem. We have a large number of talented programmers, but most applications still have to be written from scratch. There are some attempts to customise Western products, but until Polish regulations become more 'European' it will remain a Sisyphean task.

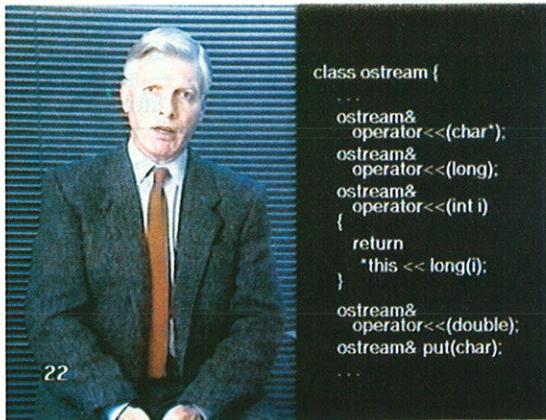
Recently, some Polish software companies have entered into partnerships with Western developers. This kind of cooperation should bring mutual benefits. First, it can help to solve the problems with the lack of manpower in some Western countries. Second, it may cut down the development costs up to five times. Finally, it may help to decrease the technological gap between West and East.

EXE

Richard Sadowski has run a private computer company 'Studio Uslug Komputerowych SAMBA' since 1986. The company is both a computer dealer and software house. Notable products include a suite of programs for civil and mechanical engineering, and it is working on various business applications and CASE products intended for a wider market. The author can be reached on 010 4858 217088 or fax 010 4858 204940.

VIDEO COURSE

Learn C++



Learn C++ Now!

The great exodus of programmers from C to C++ has begun! Since C++ builds on C, it's the easiest OOP language to learn. That's why it's called "the language of the 90's".

Why the rush?

PRODUCTIVITY! Yes, C++ programmers can write programs in less time requiring less maintenance. Large projects become much easier to manage.

Unfortunately, learning C++ can be very costly. Classroom instruction is expensive even without the travel and hotel costs. Of course, not learning will cost you even more in the long run. Now there is an alternative!

The top C++ video tutorial at the lowest possible price.

The C++ video tutorial from Zortech is the ultimate C++ training tool for work or home at only £299.95. It comes on six VHS video tapes containing 32 clear, extensive tutorials.

Used in conjunction with the concise workbook and tutorial disk, you will find everything you need for fast-track C++ tuition.

The course is generic (i.e. compiler and hardware independent) and is available with or without the award winning Zortech C++ Compiler for MS-DOS and OS/2.

Start writing C++ code within a week.
As a C programmer, you will start producing C++ code within a week of concentrated use of this course. Alternatively, spend just an hour a day watching the video and working through the suggested exercises to learn C++ in only six weeks!

The Leader in video tutorials.

If you don't already know C, you can join tens of thousands of programmers world-wide who have learnt C with the Zortech Complete C Video Course, described as:

**NO RUSH!
NO TRAVEL!
NO HOTELS!
ONLY £299.95**

"An excellent bargain ...
... I heartily recommend!"
Gary Ray, PC WEEK

Save your company thousands of £££'s.

One programmer can train for only £299.95, but you can train ten programmers for just:

C++ Course £299.95
9 Extra Workbooks £179.55
Total 10 students £479.50

Yes! Only £47.95 each!
(With all the FREE refresher courses you need!)

Now, Zortech with its new C++ Video Tutorial has refined the art of video tuition and presentation even further for the 1990's.

IN EUROPE CALL:

44- 1-316-7777

ORDER FORM

Please rush me these items:

Qty	Description	Price
—	C++ Video Tutorial	£299.95
—	Extra C++ Workbooks	£. 19.95
—	Zortech C++ Compiler	£129.95
—	Developer's Edition	£299.95

Please add £5 + VAT postage in UK
All UK orders please add 15% VAT
International shipping charged at cost

Name _____

Company _____

Address _____

Phone _____

MC,VISA or CHQ _____ Expiry _____

Card No. _____

"With or without a mouse, it's the cat's whiskers."

New development environments and technology transition make 80's editors inefficient and obsolete. To be competitive, programmers must have leading edge tools. The tool you'll use the most, the one that wraps everything into a single consistent environment will be the powerful, configurable SPE Professional Editor.

Right out of the box, it'll make you more productive, more adaptable, more imaginative. Its instant installation, elegant mouse support, advanced user interface, and point-and-shoot help get you running immediately.

MS-DOS, OS/2 and Dual Mode together

MS-DOS, OS/2 and Dual Mode versions are all included in the same

box on both 5 $\frac{1}{4}$ " and 3 $\frac{1}{2}$ " disks, with templates for popular languages, with or without a bundled Microsoft® Mouse. Great value, and your team can move into the 90's with the same editor.

Turnkey emulations

If you prefer the commands and keystrokes of a popular editor, our turnkey emulations duplicate them precisely. And you still gain the

SPE Professional Editor's advanced features, windowing capabilities and powerful engine.

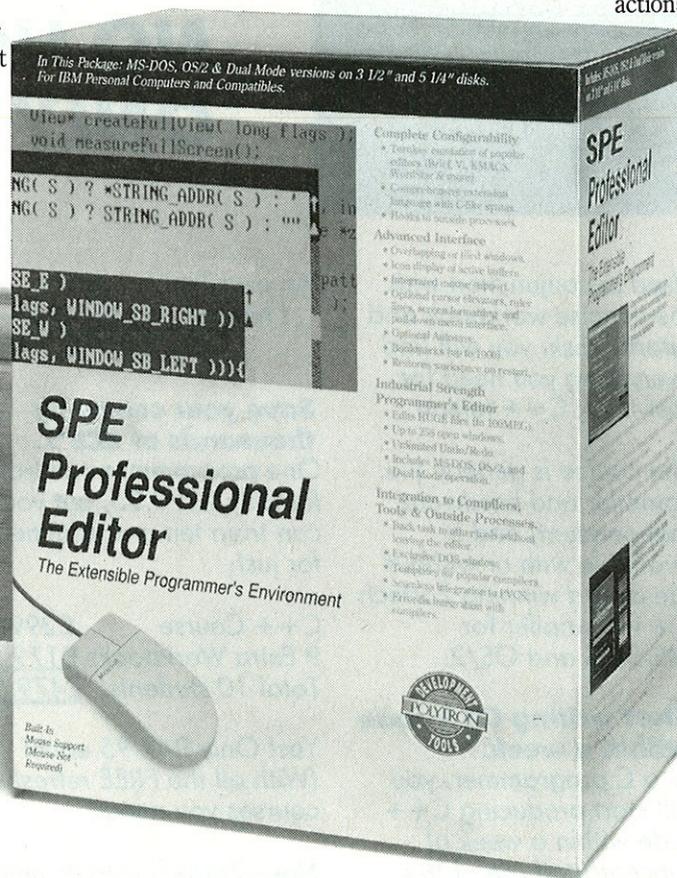
With or without a mouse

Use the editor with or without a mouse — all functions are available without lifting your fingers from the keyboard. But the click-select windows, scroll bars, zoom, shrink-to-icon, block text selects, and other speedy mouse actions make editing extremely efficient.

Don't take our word for it

We're confident that the SPE Professional Editor is the world's best programmer's editor. But don't take our word for it.

Order your copy and decide for yourself.



Works with or without a mouse.
Packaged with or without Microsoft® Mouse.

Version 1.1 now available

APPROVED DEALERS

Software Paradise
Avenue House, King Edward Avenue
Caerphilly Mid Glamorgan CF8 1HE
Tel: (0222) 887521

Computer Solutions Limited
Canada Road, Byfleet
Surrey KT14 7HQ
Tel: (0932) 52744

Grey Matter Limited
Prigg Meadow, Ashburton
Devon TQ13 7DF
Tel: (0364) 53499

Roundhill Computer Systems Limited

Orchard House
Ogbourne St. George,
Marlborough, Wiltshire SN8 1SU
Tel: (0672) 84535

Reflex Technology Limited

9 Buckingham Place
Bellfield Road, High Wycombe
Bucks HP13 5HW
Tel: (0494) 465907

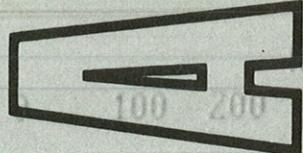
Frontline Distribution Limited
Intec 1, Wade Road, Basingstoke
Hampshire RG24 0NE
Tel: (0256) 463344

System Science
3-5 Cynthia Street
London N1 9JF
Tel: 071-833 1022

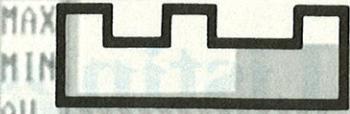
Software Express
Portmill House, Portmill Lane,
Hitchin, Herts SG5 1DJ
Tel: (0462) 422525

File Run Windows Analysis Display Memory Configure SEA

FUNCTION

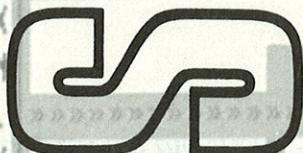


Brakes_on



SYSTEM EXECUTION ANALYSIS
for 8051 and 68HC11
embedded systems designers.

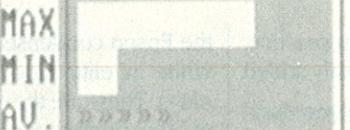
Fuel_Inject



Speed_Increas



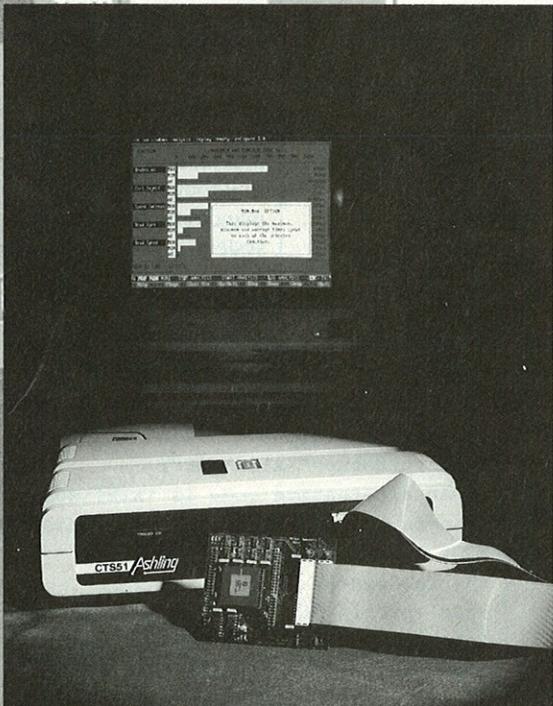
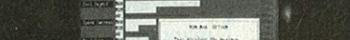
Read_Gyro



Read_Speed



ELAPSED TIME



Now available for 80C196

SEA equips you with a secure and accurate method of validating your system's performance characteristics. You can:

- Use it to verify and record your system's interrupt service response times and interrupt service frequency.
- Identify which program functions are called most frequently and which ones take the most time.
- Isolate situations where performance specs are violated or are marginal.
- Timestamp function call sequences and optimise your logic structure.
- Locate redundant, unused, or untested code and variables.

Finally, you can LOG to disk and print a report verifying that you have tested for and have met your performance and response specifications.

Physical Description:

The SEA is a plug-in card for the in-circuit emulator of Ashling's CT-series Microprocessor Development Systems. The SEA uses a high speed DSP device to measure continuously, in real-time, the following parameters for up to 126 functions:

- Percentage of overall time
- Number of occurrences
- Minimum and maximum execution times
- Trace and Timestamp of function flow (8K Events Deep)

The SEA is fully integrated into Ashling's high level debugger, PathFinder, and Ashling's ViewFinder Windowing interface by means of a pull-down menu. It can be accessed and made operational within seconds and without interrupting emulation.

See us on Stand No 26
MDS, Wembley Conference Centre
Feb 5th-7th 1991.

Telephone or fax now for Interactive Demo Disk,
On-site Demo or further information:

Ashling
The Development Systems Company

Ashling Microsystems Limited
Butler House
19-23 Market Street
Maidenhead, Berkshire,
SL6 8AA U.K.
Tel: (0628) 773070 Fax: (0628) 773009

Two Column Listings

Dan O'Brien presents a plan to save the rain-forests via an undocumented feature of some Epson compatible printers.

It has always mildly annoyed me that printed program listings never quite reach the end of the line. Like poetry - terribly wasteful and namby-pamby. You'd never catch a hex dump missing valuable space like that. Eighty-column documentation grates, too, when it's sprawled across an A4 size page. So, in a spirit of recession-combatting, tighten-our-belts eco-thriftiness, this month's Code Page presents a short C program to print out programs and documentation in two A5 columns of 80 characters each.

According to dogma, this is impossible on strictly Epson compatible printers, which can only run 132 columns in condensed pica mode. Fortunately, what is impossible

in theory is entirely reasonable in practice, as 160 columns is generally an easily added

The program sacrifices speed for readability, I'm afraid

feature for compatible manufacturers to implement. The actual control codes vary between brands; the most common is to use

the Epson condensed text code (ASCII 15) while in elite mode (activated with ESC <M>). These are the codes used here. Users of other printers can discover their own 160 column mode commands through a cursory manual browse.

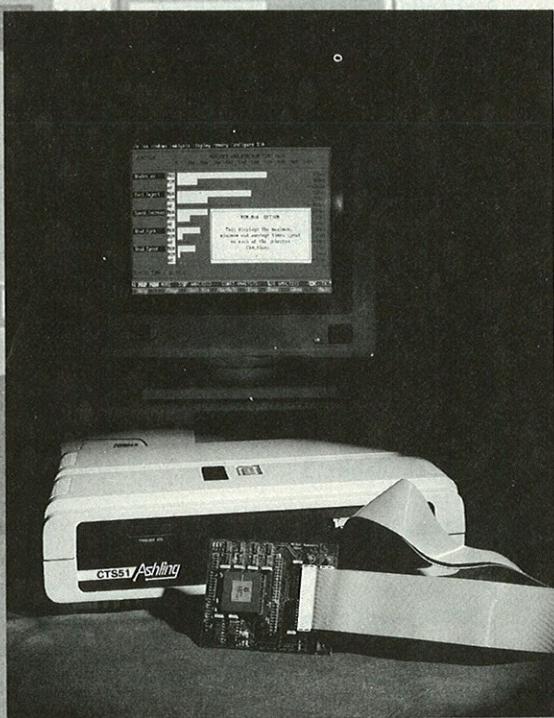
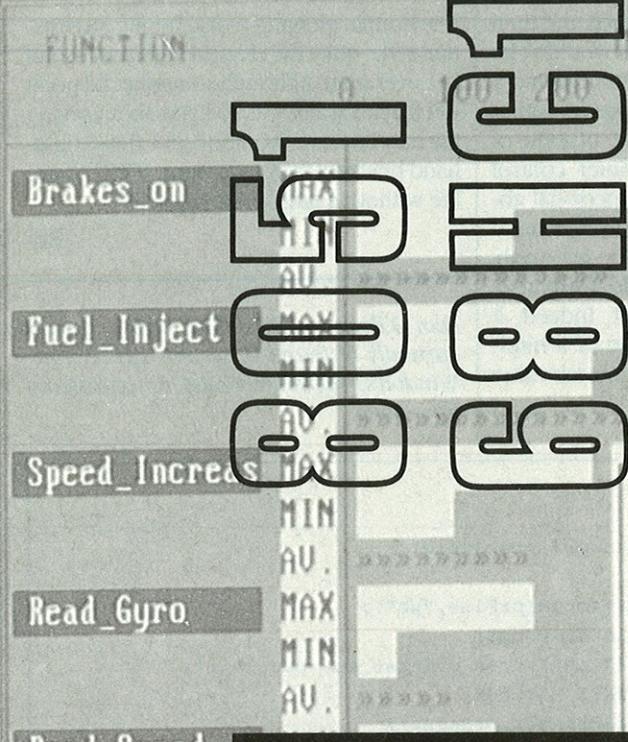
The program itself is a cinch. As you would expect, it uses a buffer to read in text a page at a time, rearranging the text as it prints. The toughest code is reading in the command options. It should run on any ANSI C system, although you may have to implement your own version of strdup if your libraries do not supply it. Watch out for the 't' option in the input file fopen - that 't' isn't ANSI, but many PC compilers need the

```
/* PR2- Two column printer */
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <ctype.h>
#define TRUE 1
#define FALSE 0
typedef int BOOL;
/* Procedures: */
void setupPrinter(); /*initialise output stream */
void printOut(char *); /*print single file*/
BOOL loadColumn(int,FILE *); /*read one column */
void expandTabs(char *);
void overflow(char *); /*trap overlong lines*/
void printPage(void); /*print full page*/
char *skpspace(char *); /*skip white space*/
void xerr(char *,int); /*error handler*/
/* Non-ANSI functions: */
char *strdup(char *c); /* gets malloced copy*/
#define MAXLINELEN 1024
#define MAXPAGESIZE 100
#define CWIDTH 80 /* Width of column */
/* Printer codes /
char *prFF="\014";
```

```
char *prSeparator="\10|";
char *prSetup="\x1BM\x0F";
/* An alternative code is \x1B!\x05 */
/* default values for all the options...*/
BOOL separate=FALSE;
int tabSize=8;
char *prFile="LPT1:";
int pageSize=60;
char page[2][MAXPAGESIZE][CWIDTH+1];
/* emptyStr should have CWIDTH spaces in it */
char *emptyStr=" ";
FILE *printer;
int main(int argc, char *argv[])
{
char *optarg;
int optind=1;
if (argc==1)
{
xerr(" \
Usage:pr2 [-s] [-o <outfile>] [-t <tabsize>]
[-l <length of page>]\ \
<fname>\n"
,EXIT_SUCCESS);
}
```

Figure 1 - The Pr2.c listing

File Run Windows Analysis Display Memory Configure SEA



CT SERIES MICROCONTROLLER

DEVELOPMENT SYSTEMS

for 8051 and 68HC11 embedded systems designers

Ashling Microsystems has set the standard for high-quality Microcontroller Development Systems since 1984 and continues to provide the outstanding package of features. In particular, Ashling's latest CT-Series products provide you with unique capabilities that will help you deliver on your design project targets.

Management of Variables: The Ashling "PathFinder" High Level Language Debugger can interpret the most complex array and structure variables. Out-of-bounds array indices are detected and flagged. The Variables Monitoring Window is updated to reflect the current contents of the simple type, the specific complex array element, or the individual structure member referenced. Expressions involving variables of simple and complex type can be formed and are updated in the variables window.

Automatic Synchronisation of Debug Windows: PathFinder links your Software and Hardware painlessly and efficiently by automatically synchronising the following windows: Original Source Code; Disassembled Memory; Traced Source Code; Traced Bus Activity; and Traced I/O Port Activity.

System Performance Verification: The CT series can provide a continuous, real-time measurement and log of system responses and can verify that you have tested for and met the performance specifications of your design. You can determine (to 2 microsecond accuracy); interrupt service times; frequency of interrupt occurrence; elapsed time between interrupts; system execution time distribution; and timestamped function flow trace.

Efficient Operating Environment: You can move quickly from one development, debug, test and validate function to the other. The Ashling Development Environment gives you, Software Development; High-Level Language Debug; In-circuit Emulation; Real-Time Trace; System Performance Verification; PROM Programming; On-Line cross referenced HELP; and External File Browse; all seamlessly integrated in a single debug environment.

Cost Effectiveness: The CT-series acquisition options are also flexible and customer requirement orientated. System prices start at £ 2,999 with a range of modular, inexpensive upgrade paths. There are Rental, Rental-to-Purchase, Purchase and Lease Plans; while Renewable Warranty and Software Update Plans provide control of maintenance costs and protection of your investment into the future.

See us on Stand No 26
MDS, Wembley Conference Centre
Feb 5th-7th 1991.

Telephone or fax now for Interactive Demo Disk,
On-site Demo or further information:

Ashling
The Development Systems Company

Ashling Microsystems Limited

Butler House
19-23 Market Street
Maidenhead, Berkshire,
SL6 8AA U.K.

Tel: (0628) 773070 Fax: (0628) 773009

CIRCLE NO. 503

text mode option to be made explicit, as it involves some data translation.

The form of the program sacrifices speed for readability, I'm afraid. The data structure for text, for instance, should make it easy to add more columns. This is a working program in the sense that I've been using it for some years, so there are a couple of 'features' which I couldn't bring myself to rewrite. The most glaring is the static declaration of the buffer arrays, and the huge string allocations. I instinctively shy clear of malloc, and when I do use it,

tend to snatch huge lumps of KB and then hoard it. In an ideal world, this wouldn't be necessary; at present, I have more memory than I do debugging patience. Another kludge is the hard-wired nature of some of the options, notably the printer control codes. Again, this is due to a personal abhorrence of huge option lists in command lines, and a financial tendency to have only one printer. Finally, PR2 should, I suppose, wrap-around overly long text. Indeed, it used to, but I found this more of a hindrance than a help in listings. A low tab value is a better solution.

The normal glowing praise for my indentation style: quite the clearest and most useful I've ever seen. Although someone did point out that if I really was so keen about saving the rain-forests, I wouldn't give those right-hand brackets their own lines. Well, what's life without a little extravagance?

EXE

Dan O'Brien is .EXE's News Editor. He can currently be found in section 12-24 of the Windows Programmer's Reference, lost and very alone.

```

for (*optarg=skpspace(argv[optind]))=='-' ;
optind++;
{
switch (*++optarg)
{
case 's':
separate=TRUE;
break;
case 'o':
if (++optind!=argc)
prFile=strdup(argv[optind]);
else xerr("What output file?\n",
EXIT_FAILURE);
break;
case 't':
if (++optind!=argc)
tabSize=atoi(argv[optind]);
else xerr("What tab size?\n",
EXIT_FAILURE);
if ((tabSize<0) || (tabSize>CWIDTH))
xerr("Silly tabsize\n",
EXIT_FAILURE);
break;
case 'l':
if (++optind!=argc)
pageSize=atoi(argv[optind]);
else xerr("What page length?\n",
EXIT_FAILURE);
if (pageSize>MAXPAGESIZE)
xerr("Page size too big!\n",
EXIT_FAILURE);
break;
default: xerr("Bad option\n", EXIT_FAILURE);
}
};

setupPrinter();
if (optind==argc)
xerr("What input file?\n", EXIT_FAILURE);
while (optind<argc)
printOut(argv[optind++]);
fclose(prFile);
return(EXIT_SUCCESS);
}

char *skpspace(char *i)
{
while ((isspace(*i)) && (*i)) i++;
return i;
}

void setupPrinter(void)
{
printer=fopen(prFile, "wb");
if (printer==NULL)
xerr("Can't open printer for output\n",
EXIT_FAILURE);
fprintf(printer, prSetup);
}

void printOut(char *fname)
{
FILE *inFile;
BOOL atEnd;
inFile=fopen(fname, "rt");
if (inFile==NULL)
xerr("Error: Couldn't open input file\n",
EXIT_FAILURE);
atEnd=feof(inFile);
while (!atEnd)
{
atEnd=loadColumn(0, inFile);
if (!atEnd) atEnd=loadColumn(1, inFile);
printPage();
}
fclose(inFile);
}

BOOL loadColumn(int pageNum, FILE *inFile)
{
char (*currPage)[MAXPAGESIZE][CWIDTH+1];
BOOL fin=FALSE;
int lineNum=0;
currPage=&page[pageNum];
for(;((lineNum<pageSize) && !fin);lineNum++)
{
char currLine[MAXLINELEN];
int currLen;
char *tmpPnt;
if (fgets(currLine, MAXLINELEN, inFile)==NULL)
fin=TRUE;
else
{
expandTabs(currLine);
if ((tmpPnt=strchr(currLine, '\n'))!=NULL)
*tmpPnt='\0';
}
}
}

```

Figure 1 - The Pr2.c listing (continued)

G.W. Computers Inc.
ADCI, 4 Eagle Square.
E.Boston. MA02128
U.S.A.

Tel: **617-569-5990**
Fax: **617-567-2981**

Xvale Ltd., Store 'D'
55, Bedford Court Mansions,
Bedford Avenue, London.
WC1B 3AD U.K.

Tel: **071-636-8210**
Fax: **071-255-1038**

Menu-IX System-IX DBMS-IX Sysix AI Spell Merge Lock GURU Crypt Tawk Tgrep Cut More Where Term Ctl Sort Grep Idle GB Trees Evaluate C Source Banner Help Head Tail Compaq Toshiba IBM Epson Qume NEC Seagate Wyse Relysis Zenith Priam Ranger Xenix Xunix Net Unix Interactive System V TCP/IP DOS Token-Ring OS/2 Ethernet Novell LAN Mic Net Aix VP-IX VMS QMS

We sell Hundreds of brand name DOS and UNIX software, and offer wide in depth experience on networks. CALL US!!.

LINX SYSTEMS STANDARD FEATURES

1 * 5-1/4 " drive
1 * 3-1/2 " drive
Dual HD/FD Controller
101 Key Keyboard
Mini - tower case
2 Serial/1 Parallel/1 Game
Built in Diag / AMI bios
Mono Screen / Adapter
CPU/Mhz+Ram = Price

8088 /10+640k	£545
80286/12+2mb	£745
80286/16+2mb	£825
80386/16+4mb	£1145
80386/20+4mb	£1345
80386/25+4mb	£1495
80386/33+4mb	£1745
80486/25+8mb	£2995

**** For SVGA Colour****
Add Screen/Card £245.00
Then just add the
HARD-DISK of your choice



Hard-Disks

20	£150.00
40	£195.00
80	£325.00
100	£495.00
120	£595.00
170	£695.00
300	£1095.00
600	£1495.00
Ethernet	
wd8bit	£135
wd16bit	£155
lnx8bit	£110
lnx16bit	£135

INTERACTIVE R 2.2

UNIX PLATFORM	
Application	2 usr
Application	3+usr
Network	2 usr
Network	3+usr
Workstation	2 usr
Workstation	3+usr
UNIX DEVELOPER	
Application	2usr
Application	3+usr
Network	2usr
Network	3+usr
Workstation	2usr
Workstation	3+usr

SCO

Unix 4 port serial	£145
Unix 8 port serial	£195
Intelligent 4 serial	£395
Genius mouse 3but	£45
HOT ITEMS	
Guru reprter	£49.95
Syseditor	£49.95
Cshell	£49.95
Menuix	£49.95
Sysix	£99.00
Sysix Pro	£145.00
SystemIX	£295.00
PC-Connect	£450.00
Pc-interface	£170.00
PC-TCP	£375.00
PC-Xview	£245.00
Terminl ctl	£695.00
Xvision	£295.00
Coherent ux	£99.00
DosBridge	£295.00

*** BUNDLED SYSTEMS include software Value up to £1730 FREE ***

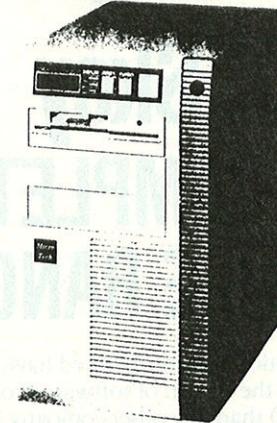
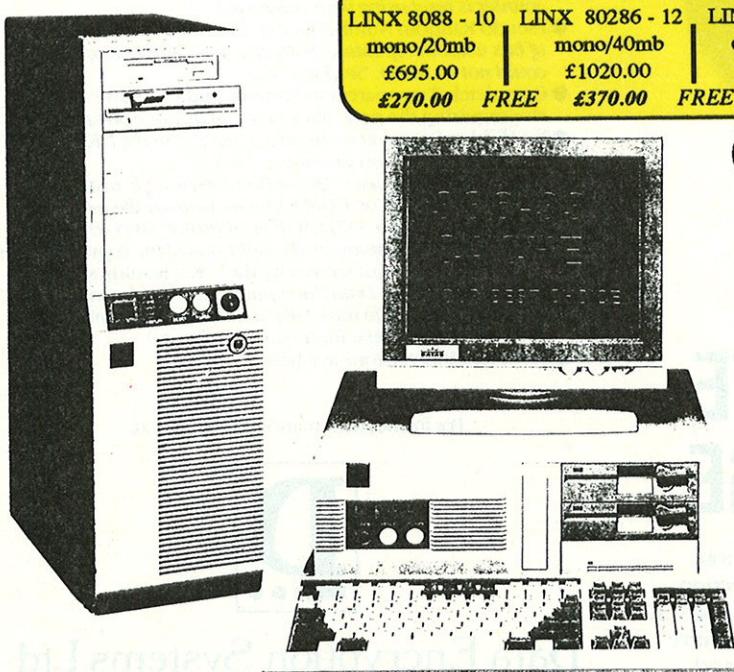
Buy a 386/33 mhz with SVGA screen and 120 mbyte h-disk at £2585.00 , get * INTERACTIVE UNIX Sys V, Rel 3.2, V2.02 * RUNTIME / VPIX and SOFTWARE DEVELOPMENT + MENUIX OR RUNTIME / TCP-IP and XWINDOWS + MENUIX

FREE !. CALL for latest deals. (24hr FAX)

** UNBEATABLE SYSTEM BUNDLES includes **FREE SOFTWARE** **

LINX 8088 - 10	LINX 80286 - 12	LINX 80386sx-16	LINX 80386-33	LINX 80486-25
mono/20mb	mono/40mb	colour/80mb	colour/120mb	colour/170mb
£695.00	£1020.00	£1760.00	£2535.00	£4360.00
£270.00	FREE	£370.00	FREE	£1300.00 FREE
		£800.00		£1730.00

CALL FOR DETAILED SPECIFICATIONS
and any special offers



Limited time offer on any item of software IN HOUSE while stocks last !!. Basic deals are intended to be 386 system UNIX SYSV, REL3.2,Ver2.02 Workstations CALL NOW and reserve yours !.

```

if (strlen(currLine)>=CWIDTH)
    overflow(currLine);
    if (strchr(currLine,'\'14')!=NULL) break;
    strncpy((*currPage)[lineNum],
    currLine,CWIDTH+1);
}
};

return fin;
}

void expandTabs(char *currLine)
{
char tmpLine[MAXLINELEN];
char *tmpPnt=tmpLine;
char *currPnt=currLine;
while (*currPnt)
{
    if (*currPnt=='\t')
    {
        int sizeOfTab;
        sizeOfTab=tabSize-((tmpPnt-tmpLine)%tabSize);
        for(;(sizeOfTab>0);sizeOfTab--) *tmpPnt++=' ';
        currPnt++;
    }
    else *tmpPnt++=*currPnt++;
}
*tmpPnt=0;
strcpy(currLine,tmpLine);
};

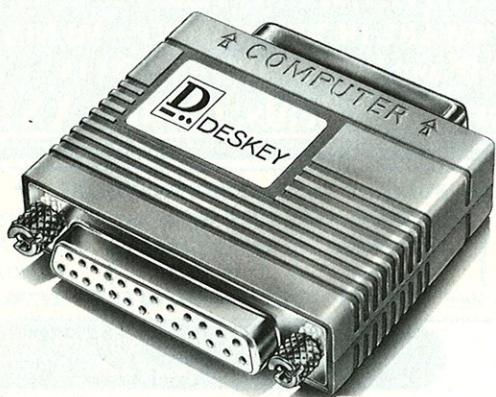
void overflow(char line[])
{
{
fprintf(stderr,"Line too long!\n%s\n",line);
fprintf(stderr,"Chopping to %d chars\n",CWIDTH);
line[CWIDTH]='\0';
};

void printPage(void)
{
int lineNum;
for(lineNum=0;lineNum<pageSize;lineNum++)
{
    strncat(page[0][lineNum],emptyStr,
            (CWIDTH-strlen(page[0][lineNum])));
    strncat(page[1][lineNum],emptyStr,
            (CWIDTH-strlen(page[1][lineNum])));
    fprintf(printer,"%s",page[0][lineNum]);
    if (separate)
        fprintf(printer,"%s",prSeparator);
    fprintf(printer,"%s\n",page[1][lineNum]);
    page[0][lineNum][0]='\0';
    page[1][lineNum][0]='\0';
}
fprintf(printer,prFF);
}

void xerr(char *s, int r)
{
fprintf(stderr,s);
exit(r);
};

```

Figure 1 - The Pr2.c listing (continued)



DESkew THE COMPLETE DONGLE RANGE

Data Encryption Systems Limited have more experience in the design of software protection devices (dongles) than any other company in the U.K.

12 years experience in the design of dongles have gone into our latest 2 products: The DK1 and The DK12, both of which use the very latest in ASIC technology.

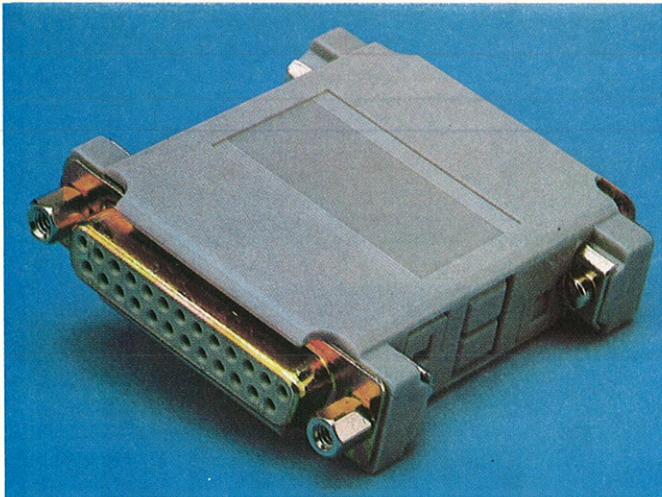
- 128 BYTES of memory split into 2×64 . Half is intended for both read and write, the other half is read-only. Writing to this protected half can only be performed using your unique password. This feature can be used for tasks such as identifying the modules of a multi-module package which have been paid for.
- Down Counter. This enables a customer to be sold 'n' goes of the software. After which the DK1 stops working until the counter is reset using your password.
- Pseudo Random Number Generator. Billions and billions of bits without repeating! Software and Data Encryption could not be easier. 'Seedable' too.
- Completely Transparent to Printers, End Users and anything else requiring the port. 16 or more on any one parallel port.
- No Hidden Extras. Everything you need is in the DK1. No need for extra 'programming' units.
- DESlock Software uses DK1 or DK12 to provide instant encryption of .EXE or .COM software without the need for access to source (or .OBJ). Will even protect DOS commands. Keeps you/your employees the right side of the copyright laws.
- Designed and manufactured by the U.K.'s leading software security specialists Data Encryption Systems Limited.
- DK12 is a reduced feature DK1, available at a quantity price to beat ALL the competition (ask for a quote).
- Serial and Bus versions available.

For further information, contact us at,



Data Encryption Systems Ltd

Edbrook House, Cannington, Bridgwater, Somerset. TA5 2QE
Telephone (0278) 653456 Fax (0278) 653300



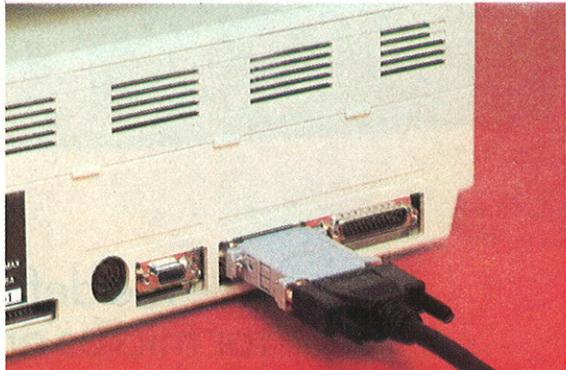
How many users of your software paid for it???

SOFTLoK International Limited was established in 1987 with the introduction of our SOFTLoK and SOFTLoK PLUS devices to combat the ever increasing problem of software piracy. Our SOFTLoK range of software protection devices are used by hundreds of software developers from small consultancies to large multinationals. To cope with the ever changing needs of our customers we have developed SOFTLoK II which combines the programmable features of SOFTLoK PLUS with a low unit cost similar to our original SOFTLoK product.

SOFTLoK IITM

The Next Generation

SOFTLoK II units are programmable devices containing read/write memory protected by a password. Both the memory and the password can be changed at any time using our routines in your application software. Easy to use menu-driven software is provided to allow small or large batches of SOFTLoK II units to be programmed with their initial data & passwords ready to be sent out with the protected software product.



As SOFTLoK II units plug into the parallel printer port they can be installed or removed in seconds.

Price: 1-19 £16.90, 20-49 £15.40, 50-99 £13.70, 100 + £11.80, 500 POA

Evaluation kit £20 (SOFTLoK II, manual & software)

All prices excl. VAT and delivery

- For IBM PC, PS/2 and compatibles
- Uses parallel printer port
- Totally transparent to printer
- Secure data & password can be changed from your application software
- Cascadeable
- 240 bytes of secure read/write memory
- 8 byte (64 bit) password
- No programming adaptors required
- Easy to use SOFTLoK II setup software
- Routines ready to link with various compilers
- Easy to follow manual

SOFTLoK International Limited

Units 10 & 11, Railway Road Industrial Estate,
Darwen, Lancashire, BB3 3EH, England.
Tel (0254) 772220 Fax (0254) 705956



The definitive database for micros and micro-networks.

MAIN FEATURES

- Clear Concise English-like programming language.
- Rapid system development and easy maintenance.
- Relational and Hierarchical structures.
- Variable length records.
- Powerful facilities for reading and writing files from other systems - e.g. Pegasus, Lotus 123.
- Run-time licences available.

OS/2, PC-DOS, MS-DOS, XENIX,
CDOS, MS-NET, NOVELL ETC.

Microft Technology Limited
The Old Powerhouse, Kew Gardens Station
Kew, Surrey TW9 3PS.
Telephone: 081-948 8255

£560+VAT Single User.
£950+VAT Network Version up to 5 Users.
£100+VAT each subsequent User.

CIRCLE NO. 507

Books

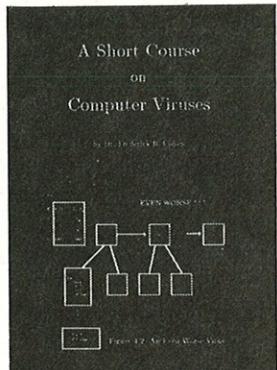
Tomes that look at viruses and data structures.

See It, Kill It, Get Away Quickly

Dr Frederick B Cohen has been writing and speaking on viruses since they were just glimmerings in the eyes of perverse minded computer scientists. And, as a perverse minded computer scientist himself, Dr Cohen does know his stuff. He was responsible for one of the earliest academic experiments in virus construction, a 200 line UNIX C program written in 1983, which took out the university computer in five minutes. Cohen was, in his own words, 'somewhat surprised' at the success of the program and has since spent his time trying to convince others about the dangers of viral attack. Not many people listened to Cohen in 1983, but they certainly do now. *A Short Course In Computer Viruses* is Cohen's way of reminding us that He Told Us This Would Happen.

The book itself is a distillate of a one day course Cohen holds and is consequently technical in content while chummy in style. Longer jokes Cohen has moved for continuity's sake to a Funny Appendix; anecdotes are mostly taken from Cohen's Wilderness Years. As far as technical details are concerned, *A Short Course* takes the reader from a brief outline of the theory of viral infection, and from this goes on to demonstrate exhaustively how no computer system can be entirely protected from viral attack. Cohen then demonstrates that most computer systems (even supposedly high security installations, like the Bell-LaPadula security system) are not only vulnerable to viral infection - they are practically begging for it. In Bell-LaPadula, for example, the rule is that you can't read information higher than your security level, and you can't write (leak) information to humbler security levels. It was very popular with the US National Security Agency for a time. Unfortunately, people are nosy, particularly security agency-type boss-people, and (as Cohen points out) access to lower levels is just an invitation to run underling programs, including, of course, any viral programs sneaked in by the mysterious temp with the central European accent.

After dealing with these systems, it's a fairly simple job for Cohen to document the disasters so far, which he does with some glee. As well as high security systems, he also gives some background information to many of the more infamous infestations of the last few years. Also listed are a few theoretical attacks: interesting virus types that (probably) don't exist, but are at least possible. Included



are intriguing descriptions of a spreadsheet macro virus, and a very odd tale about Ken Thompson, the man who co-wrote UNIX. Not wishing to pirate all of Cohen's best stories, I won't give away the details, but if Ken wasn't joking, UNIX sysops world-wide can rest easy in the knowledge that at least one of their account-holders knows how to work vi.

Cohen is ultimately cynical about any form of protection, but this does not stop him from going into a detailed examination of current approaches. Part two of the book is devoted to devising practical defences. Technical methods - virus protection programs - he divides into three categories: sound defences that will always limit viruses (though usually with unpleasant side-effects), solid defences which will usually work, given some care, and rubbish defences, which belong in the bin. Interestingly, the one defence which Cohen has no time for at all is virus scanning programs, which he dismisses with some well aimed swipes.

The chapter on non-technical defences takes a broad interpretation of the word 'defence', including as it does techniques for recovering from viral attack (three sections called See it, Kill it, and Get Away Quickly) and how to trace the miscreant.

Cohen will still occasionally mention how it won't do any good anyway, and how we are all doomed. But he does know when to tone down the heavy kismet, and become more practical: chapters 8 and 9 explain how to perform a cost analysis of viral defences, together with a list of approaches for small business, university, financial institutions and computer company installations. Finally, to wrap it all up, there is an annotated bibliography of the subject, including most of the landmark papers.

It seems pretty conclusive, and is certainly an entertaining read if you enjoy laughing out loud at the poor misfortunes of others, as I do. Provisos? Well, Cohen does have a few axes to grind, and his belittling of other anti-virus utilities might have something to do with his academic stance as promoter of Integrity Shells (which are, by contrast, heavily plugged as a defence). The mathematics which Cohen occasionally falls back on isn't always necessary, and is occasionally used, it seemed to me, just to make it all seem terribly definitive and clever when it's not. Also, the book is only 200 pages of thinly-spaced text, which is a little dinky for twenty five quid. Still, Cohen does have clout, and is not as self-appointed a viral expert as most. If you're considering how best to defend yourself from the viral lurgy, you could do considerably worse.

Title: A Short Course on Computer Viruses

Price: £25.00

Author: Dr Frederick B Cohen

Publisher: ASP Press (In UK, contact IDS on 071 631 0548)

ISBN: 1-878109-01-4 Pages: 200

Books Received This Month

- Hypertext and Hypermedia: Theory and Applications* by Nigel Woodhead
- C for FORTRAN Programmers* by T.D. Brown Jnr
- The Programmers Guide to DB2* by Paul H. Byall
- Object-Oriented Languages, Systems and Applications* by Blair, Gallagher, Hutchinson, Shepherd
- The Practical Guide for People With A New Idea* by Laurence Shaw

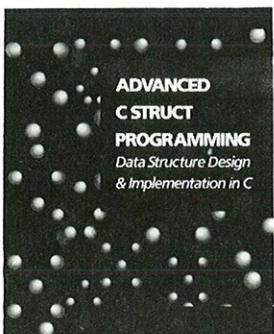
Addison Wesley	£19.95	ISBN 0-20-154442-3	pp400
Prentice-Hall	£17.95	ISBN 0-13-120528-5	pp206
John Wiley	£24.50	ISBN 0-471-52227-9	pp240
Pitman	£24.95	ISBN 0-273-03132-5	pp370
Patent Eye	£9.95	ISBN 0-950-8559-01X	pp150

Oceans, webs, spells and structs

The big mystery about *Advanced C Struct Programming* is the title. As its author, John W L Ogilvie is quick to explain, this is not a book concerned with just structs. It's not even just about C. Instead, it's an attempt to discover a structure construction algorithm: a technique, which, if routinely followed, should establish all of the requirements of a data structure before its implementation.

It's a tall order. Ogilvie thinks he knows the answer: a huge checklist of about 50 complicated queries, which, like those quizzes in Reader's Digest, reveals all about your data's personality. Seem simplistic? Well, it's only the core of this hefty book, and the questions themselves are not exactly multiple-choice. Question one, for example, asks 'What is the geometry of this abstract data type?', which takes some explaining in itself. You begin to grasp the questions only after ploughing through the many worked examples.

This struct construction kit is part of a wider plan which includes almost the entire life-cycle of a program, from initial specification to debugging and maintenance. Ogilvie has questionnaires for these tasks, too. But even that's not enough; there are chapters on all kinds of levels, from specific debugging tips (including a lovely technique for making dumps readable by choosing hex values which spell out their purpose: 0xDEADC0DE for example) right



through to a very brave attempt to explain how programmers think to the managerial mind. Struggling somewhat here, Ogilvie attempts to convey the act of programming through comparison to swimming in oceans, clutching at webs and weaving spells. The metaphors sound trite, but Ogilvie's descriptions ring true. Sadly, he's trying very hard to do the nigh impossible.

Which rather sums up the book. One can see what Ogilvie is trying to present. Most programmers feel the frustration of being caught in an otherwise precise discipline with nothing but a few rules of thumb; no complete theory, and without any consistently successful approach. *C Structs* is a sort of core dump of practical knowledge, tied to an imposed structure. Ogilvie clearly thinks that his approach to programming is reflected in his set procedure. If you follow the rules, you won't go wrong. In fact, his examples show that he is often hard pressed to squeeze the instinctive approaches he does adopt, to the rigid rules he is claiming to follow.

Ogilvie is obviously an excellent programmer with a gift for communicating concepts. What he tries to do in *C Struct* is write a book that will show everyone how to program so well. Thinking like a programmer, he's convinced that if he can consistently code well, he must be following a set of rules, which he can, in turn, explain to others. This book isn't quite that explanation, but it is very near, and it's certainly one of most complete descriptions of practical programming I've read for some time. A difficult book to recommend, but an author who should be watched out for.

Title: Advanced C Struct Programming

Price: £22.95

Author: John W L Ogilvie

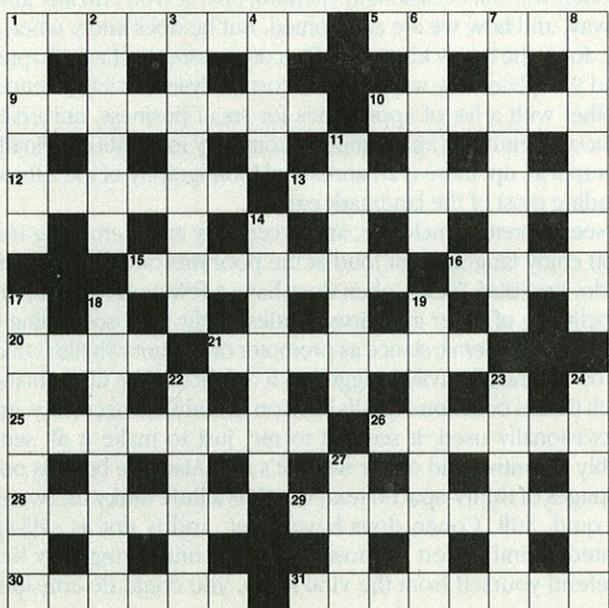
Publisher: John Wiley

ISBN: 0-471-51943-X

Pages: 410

EXE

.EXEWORLD FEBRUARY



ACROSS

- Buses for thieves in the night? (8)
- Used a 15 to step through the software (6)
- Hit rate of sporting types (8)
- Crudely thrash in scalar ruptures (6)
- Maybe deal with multiples of 2 in a fair way (6)
- Working eight bits at a time (8)
- Useful little routine! (7)
- Spill gore when you meet him in the dungeon (4)

20 Meat to software always... (4)

21 ... and label every chunk of it? (7)

25 & 26 Maybe it has a Centronics port (8, 6)

28 Young doctor may arrive when he should (6)

29 Precise splatterer when painting (8)

30 Common printers (micros?) (6)

31 Goes through it all again conservatively (8)

DOWN

- Unusual soccer shot on file (6)
- Collect 20 or rosebuds (6)
- Small analog signals produced by veal stew (8)
- Mysterious 16 of the Himalayas (4)
- 1 down loses head for king in the drive (6)
- Bearing of printer getting RETURNed (8)
- Poor layer in a diode (8)
- Hardware, software - the lot (7)
- One in a conspiracy may provide the picture (7)
- Smartly flexible in packet routing (8)
- 31 the routine (8)
- Its object is to put the code together (8)
- Once 10p to its friends (6)
- Such a display makes sense (6)
- Networks? That fits snugly (6)
- Common pointers are poor when in church (4)



.EXEWORLD DECEMBER

Editorial Index

If you are interested in the News articles on pages 4, 7 & 9 and would like to receive more information about the products mentioned, please circle the corresponding circle number on the Reader Service Card at the back of this magazine.

COMPANY	PRODUCT	CIRCLE NO
3L Ltd	Parallel FORTRAN	525
Autodesk Ltd	AutoCAD	526
Bits Per Second Ltd	Graphics Server	527
Blenheim Online Ltd	Software Tools Show	528
BSI	Business Standards	529
Cebra Communications Ltd	Video adapters	530
Cocking & Drury (S/W) Ltd	Smalltalk/V	531
Custom Micro Concepts	CU-Access	532
FAST	Pressure Group Info	533
Inmos Ltd	Transputers	534
Microsoft	OLE Specification	535
MicroWay UK	ClearView-386	536
Milspec Systems Ltd	PC RAM Utility	537
NAG	PV-WAVE:NAG	538
OMG	OMA Guide	539
OS/2 User Group	Memmon	540
P & P Micro Distrib.	dBase IV for VMS	541
QA Training	Multiscope for DOS	542
Quintec Systems Ltd	Prolog	543
STSC	APL*PLUS 10	544
Systems Resources	CASE:W for C++	545
Systemstar	C-scape	546
The Software Centre	BASIC language	547
Thornbrook	Free Ada Compiler	548
Zortech Ltd	C++ compilers	549

.EXE

The Next Quarter

- March** **Programming Library.** Libraries are one of the more useful programming tools. C++ class libraries are of growing importance — we find out what's available. Plus: the secrets of building and maintaining your own libraries.
- April** **Life After DOS.** As machines grow more powerful, MS-DOS seems set to fall behind. But what will replace it? Windows? We also investigate how to write programs which can cope with *any* operating system. Plus **The Software Training Guide**.
- May** **Graphics.** Incorporating graphics into programs is rewarding, but is often fraught with difficulties. We look at some advanced techniques, and review some of the utilities that can help.
- June** **Software Tools Show Special.**
The Software Engineers. Modern employers don't advertise for 'programmers', they ask for 'software engineers'. But what is a software engineer, and what are the tools and techniques at his disposal?

BRIC-A-BRAC, SMASH-THE-RAT, RAFFLE, CRAFT STALLS,
JUMBLE, TOMBOLA, KNOBBLY KNEES COMPETITION (3PM)
DOORS OPEN 2PM, ENTRANCE SP
PLEASE LEAVE ANY JUMBLE WITH 5TH ISLEWORTH SEA SCOUTS

POST CARD

AFFIX STAMP HERE

FRANCHISE WORK

WANTED

MUST BE ARTICULATE + TECHNICAL
Particularly looking for help with the following

- * MAY - Graphics.
- * JUNE - Software Engineering.
- * JULY - Artificial Intelligence.
- * AUGUST - FORTRAN and COBOL
- * SEPTEMBER - Communications + Networking

ALSO WANTED: FOREIGN CORRESPONDENTS
Especially for Antipodes, Africa, Near+Far East Asia, S. America

Contributors notes + payment details available on application
to: THE EDITOR, .EXE, 10 BARLEY MOW PASSAGE, LONDON W4 4PH.

STRIP GRAM

Tarzan, French waitre, Roly-poly, Policewoman, Gorilla, Champagne

Happy Half-Hour Hotel

Accommodation (Singles Doubles Twin - Bath/shower H+C water), restaurant, sun-bed. Kettle+cups+sachets of instant coffee/tea, T.V./extractor fan/trouser press for each room. Please call Mrs. Everso on the number below.

AV-PAIR / NANNY
FOR NATASHA + RUPERT
+ 8 yrs (RESPECTIVELY)
Zoom with SATELLITE TV
OF VOLVO Some WEEKENDS
CALL EDWINA SHAGGINTON - GORE
WITH FAMILY IN THE DORDOGNES
IS AVAILABLE, BUT THEN THERE
... NANNIES.

Specialist Technology Recruitment

CONSULTANCY AND PROGRAMMING FOR THE '90'S.

4GL - SUPPORT/TRAINING - HERTFORDSHIRE

TO £22,000 PA + BENEFITS

This Systems House is currently seeking a 4GL Support/Trainer. The candidate should possess at least two years experience as a 4GL analyst/programmer and should also possess classroom training expertise. This position is an unusual support role as it entails post-call systems analysis as opposed to the standard "Helpline" functions.

SENIOR CONSULTANT - HERTFORDSHIRE

TO £25,000 PA

This International Group now seeks a senior consultant possessing a minimum of 2 years PICK experience, together with a knowledge of Accountancy. UNIX expertise would be a distinct advantage. This innovative product is currently at the launch stage and therefore the successful candidate can expect genuine career progression.

INFORMIX ANALYST/PROGRAMMERS

TO £22,000 PA + CAR

Sought by a systems house looking to augment its development team. Candidates will be aged between 25 - 35 yrs., of Graduate calibre and will possess at least 2 years Informix programming experience.

AN ADABAS NATURAL ANALYST/PROGRAMMER

WEST LONDON

£20,000 PA - £23,000 PA + BENEFITS

Sought by a well known software house. The requirement is for an analyst/programmer possessing 18 months + experience gained within an IBM environment.

C/UNIX PROGRAMMERS - HERTFORDSHIRE

£21,000 - £24,000 PA

This expanding software development company is seeking several Graduate 'C' Programmers possessing a minimum of 2 years experience. Informix/SQL expertise would be an asset.

MANAGEMENT CONSULTANCY - LONDON

TO £32,000 PA + BENEFITS

This prestigious "Top Ten" firm of Management Consultants now seeks a Graduate with good consulting experience possessing a range of technical expertise covering the use of Structural Analysis/Design, CASE tools and UNIX. Use of a methodology, preferably Information Engineering, is essential. It is highly likely that the successful candidate will come from a systems analysis technical consultancy or project leadership environment.

GRADUATE RESEARCHER - LONDON

£25,000 + BENEFITS

Sought by a leading firm of consultants to provide high level computing information to their senior consultants. The requirement here is for a candidate possessing analytical and computing skills in addition to experience of report writing and the production of presentations.

C/UNIX ANALYST/PROGRAMMER - SURREY

TO £21,000 PA + COMPANY CAR.

Planned expansion has resulted in this position requiring a 'C'/UNIX Programmer of graduate calibre possessing at least 18 months experience. This position may lead to a more senior appointment.

For further information on these positions please telephone 071 839 2319 (9am-7pm) or 071 266 2848 out of office hours (until 10pm)

Specialist Technology Recruitment

94 Jermyn Street, London SW1Y 6JL

Fax No: 071-930 5977

YOUNG SOFTWARE DESIGNERS

To £20000



Leeds

Do you see answers where others see only problems???

If you are a creative software designer with such a flair, then you'll be in excellent company if you join GMAP.

GMAP custom-build Spatial Analysis and Modelling Systems that provide the senior decision makers in Times Top 1000 businesses with decision support tools many had imagined were impossible to produce.

The key to our continuing success is our policy of encouraging our people to explore new approaches to solving problems. Our customers benefit from a system that is uniquely crafted rather than a "shoe-horned" package.

We wish to recruit a number of bright, like-minded young problem solvers with the ability to think in a spatially-referenced context. You should be a person who enjoys finding your own answers rather than coding someone else's. You will need to be fluent in 'C', preferably under DOS and/or UNIX and be quick to learn other languages and operating systems, as required. Experience of graphics and database tools would be useful but not essential as training in these areas can be arranged.

Interested?? To learn more about working with **GMAP** please call **ALAN CARNELL**, our retained recruitment specialist at Concurrent Appointments for an informal discussion or mail him your CV in confidence.

INTERVIEWS WILL BE HELD LOCALLY



**Concurrent
Appointments**

Software
Recruitment

27 FIELD CLOSE
HARPENDEN
HERTS
Tel: 0582 712976
Fax: 0582 764858

Our client, Daniel Industries (UK) Limited is currently seeking to increase its software engineering staff. Based in Larbert and very well placed for road and rail connections. Daniel Industries (UK), Limited has gone from strength to strength during the 17 years it has been established at this Location.

As you would expect from a major manufacturer of flow measurement equipment for the oil, gas and petrochemical industries, they also have facilities in Aberdeen, London and the parent company headquarters in Houston, Texas.

The opportunities currently available will enable the successful applicant to pursue their career in Real Time software development and support on flow metering and control systems including development of product based software and ultrasonic metering software.

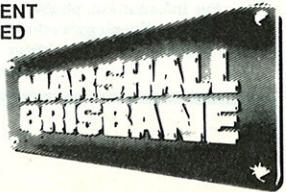
SOFTWARE ENGINEER

In order to qualify for the maximum salary of £15k per annum you will have a minimum of two years experience designing and implementing Real Time fiscal metering systems for the oil and gas industry. Your experience will have encompassed the implementation of fiscal calculations (eg ISO 5167, AEA8), with two years experience in Intel Pascal, Turbo Pascal and PLM/86 on 8086/8087 machines running RMOS. Using 8085 Assembler your ability to develop firmware for pulse interpolation and frequency measurement will have been proven.

JOHN DEMPSEY (Ref: D131)
MARSHALL BRISBANE RECRUITMENT
CONSULTANTS (GLASGOW) LIMITED
15 NEWTON PLACE
GLASGOW G3 7PY

Tel: 041 332 1396

Fax: 041 332 0974



Staff-Match

The Recruitment Consultants who Care

PC - UNIX - 'C' - SYBASE

Programmer

Buckinghamshire

Salary dep on experience

This dynamic Software House, specialising in the Financial Market, has a position for an experienced individual with flair and initiative. You will have a cheerful disposition and be working either independently or in a small team.

UNIX - INFORMIX - 'C'

Analyst/Programmer

Berkshire

Salary to £18000

An excellent opportunity has arisen for a self-motivated professional to join this fast expanding Accounting Software House. You will be working on a number of exciting new projects, developing some of the most advanced and sophisticated accounting software.

PC - DW4 - PARADOX - LOTUS

Information Centre Analyst

Wiltshire

Salary to £20500 + Excellent bens.

This well established Insurance Company is seeking an individual to supply systems software support and training to all its PC users within the company. In addition, you will also be involved in feasibility studies and promotion of IC facilities to end users.

UNIX - 'C' - ORACLE

Programmer/Analyst

Middlesex/London

Salary to £25000

We are seeking an individual for our client, a major Software House based in London. The initial assignment is located in Middlesex where you will be working on State-of-the-Art software. A minimum of 2 years' experience in the above environment is required.

PC - DATAEASE - LOTUS - NOVELL

PC Systems Analyst

Sussex

Salary to £16500 + excellent bens.

Working within the IT department of this large Insurance Company, you will provide internal consultancy to PC users. You will be expected to analyse requirements and produce (packaged) solutions to business problems. This is an excellent opportunity for a programmer looking for progression into analysis.

UNIX - COBOL

Analyst/Programmers

Berkshire/Sussex/Surrey

Salaries to £20000

We have a number of clients seeking individuals with a minimum of 18 months' solid COBOL. Good communications skills are also essential as the positions involve a great deal of client liaison.

UNIX - ADVANCED BASIC

Analyst/Programmer

Berkshire

£ Excellent salary + Car

This fast expanding Software House, who provide total business solutions, are seeking a dedicated individual to join their highly motivated development team. You will be involved in the complete life cycle of a project plus subsequent user training and support.

WE HAVE MANY OTHER VACANCIES IN THE FOLLOWING ENVIRONMENTS:

IBM * ICL * DEC * UNISYS * HP * UNIX * PRIME * PC

For information, please call Allison Barnard on 048649 595/546.

Evenings/weekends call Allison on 048649 780.

Please post CV's to: 6 The Mews, The Common, Dunsfold, Surrey GU8 4LJ
or Fax to 048649 739.

Exciting opportunities in

REAL-TIME SOFTWARE

Essex
to £25K

* TRANPUTERS

If you are a Software Professional with experience in signal processing or parallel processing then this (UNPARALLELED!) OPPORTUNITY could be for you.

Berks & Middx
to £25K

* ADA

We are currently recruiting high-calibre engineers experienced in ADA/VAX/VMS Structured Methods for a number of prestigious projects. Excellent opportunities, big company benefits.

South
to £22K

* C++

Our client, an international company providing ATE solutions is offering excellent prospects to Software Engineers with experience of UNIX/Windows/C

For further information about
these and our many other
opportunities
telephone:

0727 41101

or 0727 30602 (out of hours)

Alternatively fax your C.V. on 0727 838272

95 Victoria Street, St.Albans, Herts AL1 3TJ



CAD
CAM
CONCEPTS

Despite the recession, well-qualified technical and commercial professionals are still in demand, especially with experience in one or more of the following areas:

SOFTWARE APPLICATIONS DEVELOPMENT

GIS or finite element analysis or nc

SYSTEMS SPECIALISTS

Unix support or X-Windows/GUI tools development

COMMERCIAL SUPPORT ENGINEERS

product marketing or nc post sales support/customisation.

If you would like to know more about these or other vacancies, please send your cv to Sheila Fenlon or Michael von Rimscha, CAD CAM CONCEPTS (Recruiting), Campbell House, Campbell St, Cambridge CB1 3NE or telephone us on 0223 412965

COMPUTER PROGRAMMER: ANALYTICAL INSTRUMENTS

Polymer Laboratories Ltd is seeking an experienced programmer to join an expanding instrument development group at its Church Stretton, Shropshire, site.

Experience in 'C' programming within the MS Windows environment is required, and preference will be given to the candidate who also has experience in data analysis and instrumental control, within the Chromatography area.

The work involves the maintenance and extension of existing software packages running under MS Windows, and the origination of new programmes for instruments under development, including both control and analysis of data from such instruments.

Please contact Laura Watson for an application form.



POLYMER LABORATORIES

Essex Road, Church Stretton, Shropshire
SY6 6AX, U.K. Telephone: (0694) 723581

TEAM LEADER & ANALYST PROGRAMMER £ up to 23k

Apple Mac skills, lots and lots of development, good knowledge of 4th dimension, build your OOPS skills!! Good company with good attitude.

JUNIOR SOFTWARE ENGINEER £ up to 14k

Excellent Company, with excellent prospects, young team, buzzy environment, and a world leader, C, Assembler and Designing test software, must be fairly pernickety!!

SOFTWARE ENGINEERS & DEVELOPERS £ up to 18k

Cobol, C, OS2, Windows, Presentation Manager, all in a micro environment, your bag??? Project utilising Graphical User Interfaces, Number 1 Software House, wants pleasant 'useful', individuals, Definitely no train spotters!

LOW LEVEL GURUS 7 £ up to 24k

TASM, MASM, BIOS and recent assemblers!! Know your interrupt codes, not just the favourite ones.....a lively environment and a good sense of humour is the prerequisite, good benefits, stable company!

DEVELOPMENT PROGRAMMERS £ up to 20k

SQL Windows C and an enquiring mind with a good track record, any database application using SQL would be suitable and of course a solid MS-DOS ability.....



REGENCY
COMPUTER SERVICES

PREMIER HOUSE, 77 OXFORD STREET.
LONDON W1R 1RB FAX: 071 - 439 4972

071-287 0568 (24 HRS)

Analyst/Programmer

A Major city institution is seeking an Analyst/Programmer with experience in PICK based systems. The ideal applicant will have a good educational background, a good knowledge of one or more PICK based systems and at least 12 months' analysis experience. Future plans include a major Unix development with opportunities to cross train.

'C' Programmer

£18k

A major player in the image communications field is seeking to recruit a programmer with solid 'C' and MS-Windows experience to assist in the development of a new product. The ideal candidate will be of graduate calibre with solid development experience in 'C' under Windows. A background in image processing would be a distinct advantage.

Applications Development

Programmer

to £19k+Benefits

A division of a major financial institution is seeking a Programmer with a minimum of 2 years' solid CLIPPER development experience. The ideal candidate will be a team player with a good educational background who would view this position as a positive career move. He or she should be able to prove their involvement in either the development of financial software or the design of a large application. Additional skills such as Ingres and MS-QuickBASIC would be an advantage.

Analyst/Programmer

(Commercial Applications)

to £18k

A major software house situated in Berks is seeking an additional Analyst Programmer. The ideal applicant will have 2 to 3 years' experience developing commercial system in 'C' (and/or 4GL) under UNIX. This is a forward looking company with a variety of market places. So the successful candidate has an assured future.

8086 Assembler Programmer

to £17.5k

The market leader in the hand held market is seeking a programmer with a minimum of 2 years' experience in 8086 assembler programming to develop software for a new generation of hand held computers. The ideal candidate will be a workaholic with good experience with TASM or MASM and an understanding of PC architecture and microprocessor design. Experience of a windowing debugger or hardware emulator would be beneficial. The company is very forward looking and this is a unique opportunity to input your ideas to a listening management.

RPGII Junior Programmer

£16k to £18k AAE

A large multinational is seeking an addition to their team to assist in the development of S/36 and AS/400 (S/36 mode) prior to moving to native mode. The work is varied and may include travel both domestic and overseas. A future career progression and on-going training is assured with this company whose future is IBM orientated.

For details of these and other positions contact Terry Nelson on the number below out of office hours on 02406 5892.



RONICOM
RECRUITMENT

46-47 Pall Mall, London SW1Y 5JG
Telephone: 071-321 0245 Fax: 071-839 7629

TECHNICAL SOFTWARE

Bracknell	c£20K	to £18K
RISC ARCHITECTURE - Technically competent Software Engineers to develop new RISC ARCHITECTURES in a commercially demanding environment C, ADA, X400, SECURE SOFTWARE		Development of CASE tools and database systems for retail and food services. 1+ years of C/UNIX, Oracle SQL.
MJ01		JP04
Reading	£15-20K	£17K
HIGH SPEED COMMUNICATION project. Commercially aware software developers with C, UNIX, OSI, TCP/IP, BLISS, DEC NET.	JP01	HIGH SPEED VIDEO GRAPHICS - Software Engineers, 1 years + experience C/UNIX, system level development, 68000, Graphics Applications for TV/Broadcasting.
		MJ05
Southampton	to £25K	£18K
Speech recognition systems and advanced signal processing. Exceptionally qualified graduates with C 68000 and ISDN or CCITT7 experience to join the development arm of a small, successful technology leading company.	MJ02	Digital Exchange Developers - Software and Hardware Engineers. 2 yrs experience of C/UNIX, CCITT7, 16 bit microprocessors in a similar environment. Unlimited career potential.
		MJ04
London	to £21K	£19K
FINANCIAL APPLICATIONS - ambitious software engineers/A.P.'s with C, DOS, UNIX, WINDOWS, DB VISTA in a similar demanding environment.	JP02	ANALYST PROGRAMMER - UNIX and VMS platforms. New product development in C and Cobol.
		JP06
Woking	c£20K	£15-30K
DISTRIBUTED FILE SYSTEMS - Software engineers, device drivers, embedded processors, protocol implementation, network security, C, DOS, BIOS, WINDOWS, DATACOMMS exp.	MJ03	Protocol development - professional engineers, with C, UNIX, CASE, OSI, MAP/TOP, or TCP/IP experience to join a major player in commercial protocol development.
		MJ06
Oxford	to £18K	C17K
Smart terminals and financial systems. 1+ years experience of C, UNIX, 68000, 4GL's a bonus.	JP03	Mobile Comms - 1 years solid C/UNIX experience and the drive to succeed. Extra knowledge of OSI a bonus.
		JP05

This is just a small selection of the many current technical software vacancies we have to offer. For more information contact MIKE JENKINS and JULIE POOLE on 0442 231691 days or 0582 456417(MJ) evenings/weekends or 081 866 7019 (JP) evenings and weekends. Alternatively send your CV to Executive Recruitment Services, Hempstead House, Selden Hill, Hemel Hempstead, Herts. HP2 4LT or fax to 0442 230063

ERS

SENIOR SOFTWARE ENGINEERS

Exceptional technology demands exceptional people

To £25,000 + benefits

UNIX & 'C'

Do you pride yourself on the consistent achievement of excellence? If so, you can enjoy project responsibility from conception to completion - within a pioneering company at the very top end of a rapidly expanding market.

Here, as part of a small specialist team, you'll design knowledge-based Real Time systems. The challenge is to match their capability to the high performance racing cars they'll be linked to. Which means extracting, storing and reacting to data generated on the track - and simultaneously displaying it across a network of high resolution terminals.

To do so successfully you'll need to be results-oriented, self-motivated and technically inventive. As an individual you will also be highly committed to meeting strict timescales, and must be prepared to travel abroad occasionally. An enthusiastic UNIX or Graphics specialist, you'll already have 3+ years' experience in at least two of the following:

- **UNIX + C**
- **SCADA/Telemetry**

- **680x0 Assembler**
- **Device Drivers**

- **VME Systems**
- **Graphics**

An exclusive high tech working environment. A stimulating European culture. World leading status. These are some of the reasons why you'll find this role so attractive.

Another is the end-products themselves. Compact, yet highly sophisticated electronic management systems for the high performance automotive sector. Systems that need you to excel in software design and development.

You can look forward to an excellent salary and benefits package and, of course, impressive career prospects as our client continues to grow.

To find out more or to arrange an initial meeting, please contact **Grant Revan** or **Lorna Graham** on **(0344) 860919**, quoting ref EPF067.

Or write, enclosing a CV in the strictest confidence to:

TSI Group, The Columbia Centre, Market Street, Bracknell, Berkshire, RG12 1PA. Fax: (0344) 860581

Technology Recruitment Consultants



WEST YORKSHIRE

Programmers (particularly COBOL and 4GLs), Analyst Programmers, Software Engineers, Project Leaders, Support (pre and post sales), Trainers (MS-DOS, UNIX and Applications). Also Hardware/Software Design Engineers.

FOR YOUR NEXT CAREER MOVE AROUND WEST YORKSHIRE

Telephone Vincent Atherton on Leeds (0532) 504560 or write to:

AIREDALE RECRUITMENT

Realtex House, Micklefield Lane, Rawdon, Leeds LS19 6AX

AIREDALE RECRUITMENT

SOFTWARE DEVELOPERS OXFORD

4GLs/'C'/Databases

to £19,000

Required for highly successful manufacturer and distributor of leading edge hardware and software products. The work will involve development and implementation of commercial operating systems for use throughout this rapidly expanding organisation. Candidates should have a proven track record in applications development for networked PCs and be familiar with either Novell, OS/2 or MS-Windows. It is also essential that candidates can demonstrate a minimum of 2 years experience using one of the following development tools:- 'C', SQL, 4GLs or PC databases such as dBASE, Paradox or Dataflex.

dBASE/Clipper

to £15,000

Required for very successful software house to develop bespoke systems for nationwide client base. Candidates should have a minimum of 2 years experience of database programming in dBASE/Clipper or similar database products. It is essential that candidates possess good self-presentation skills as client visit are a feature of this important role.

To learn more about these exciting career opportunities or discuss your next career move in Oxfordshire and surrounding counties call David Adcock on Oxford (0865) 742456 & 7 or send your CV to:

Haybrook Appointments, Suite B, Regal Court,
112 London Road, Headington, Oxford, OX3 9AU

Haybrook
APPOINTMENTS



PC Recruitment Specialists

Allen-Miles

TECHNICAL RECRUITMENT

SOFTWARE ENGINEERING OPPORTUNITIES £15000 TO £22000

We have a number of clients throughout the UK with urgent requirements for Software Development and Support Engineers. The following are a very small selection:

SENIOR / PRINCIPLE SOFTWARE ENGINEERS HANTS 15 TO 22K. Leading edge Civil Avionics project. 680x0 Processors, Sun workstations 3+ years experience, in two or more of the following: 'C' / UNIX / MS-DOS / ADA / Structured Analysis / Design / C++ / Case Tools.

SENIOR SOFTWARE ENGINEERS HOME COUNTIES TO c18K. Real-Time Micro based S/W development and test. 'C' / Assembler / UNIX / MS-DOS / Logic Analysis / ICE.

SOFTWARE ENGINEERS HERTS TO c20K. Development of Integrated Office Systems. Experience in the I.T. Industry, using 'C' UNIX / DOS.

SOFTWARE ENGINEERS HOME COUNTIES TO 18K. ISDN development. Experience of Comms based Data Interfaces using 'C' UNIX / PASCAL.

SENIOR SOFTWARE ENGINEER S.COAST TO 19K. Leading small team on variety of projects. Interface companies' systems to customers hardware. Real-Time environment, using 'C'/UNIX and Assembler.

For further information, please phone ALAN THAKE or IAN COLLINS. Alternatively send your c.v. to the address below.

111a Seabourne Road • Southbourne •
Bournemouth I Dorset • BH5 2HF

TEL: (0202) 417750

SENIOR ANALYST/PROGRAMMER to £25k

Reporting to Managing Director

We need someone to consolidate and extend our technological lead and create tools to further improve our productivity. This will require:

- a thorough knowledge of UNIX/C and graphics.
- the ability to think at a strategic level.
- excellent communication skills.

Candidates must have a degree in computer science or a related subject and must be able to demonstrate a high level of achievement during several years of commercial experience.

Bourne Group Limited

We are a small successful software house producing high quality products for the corporate market, specialising in training and consultancy. We operate from modern offices close to the centre of Romsey and provide a friendly working environment.

Please apply (enclosing a CV) to:

Mr D C Fuller, Bourne Group Limited, Bourne House,
The Hundred, Romsey, Hampshire, SO51 8BY.
Tel: 0794 523301.

ADVERTISER	CIRCLE	PAGE	ADVERTISER	CIRCLE	PAGE
.EXE DIRECTORY	508	89	MICROSOFT	455	21
7x COMPUTER	473	42	MKS	485	55
ASHLING I	502	79	NEOW	489	58
ASHLING II	503	81	NU-MEGA	465	30
BITON	470	37	PINNA ELECTRONICS	496	67
BITS PER SECOND I	472	41	POLYHEDRON	476	48
BITS PER SECOND II	511	41	PROSPERO	453	18
BL SECURITY	457	24	QA	445	IFC
BLENHEIM ONLINE	467	33	QAII	488	57
BRENT COMMUNICATIONS	460	26	QA III	494	63
CEBRA	469	36	QBS	459	26
CLEARSOFT	510	OBC	REAL TIME SOFTWARE	513	24
COMSEC	492	61	RAINBOW TECHNOLOGIES	447	5
CREATIVE DATA SYSTEMS	514	36	RECITAL	495	64
CTL	491	61	ROUNDHILL	481	50
CUSTOM BUSINESS SYSTEMS	468	34	SCL	478	49
DATAFLEX	454	19	SELECT SOFTWARE	462	29
DEEPAK SAREEN ASSOCIATES	498	70	SELTEK	515	42
DES	505	84	SOFTLOK INTERNATIONAL	506	85
EXPERT SYSTEMS	479	49	SOFTWARE CONSTRUCTION COMPANY I	458	25
GENUS	466	31	SOFTWARE CONSTRUCTION COMPANY II	452	17
GIMPEL SOFTWARE	456	23	SOFTWARE GENERATION I	471	39
GREY MATTER	446	3	SOFTWARE GENERATION II	501	78
INTASOFT	490	59	SOFTWARE PARADISE	497	69
IXI	512	47	SOLUTION SYSTEMS	474	45
JENSEN & PARTNERS I	480	49	SYSTEM IX	504	83
JENSEN & PARTNERS COMPETITION	-	53	SYSTEM SCIENCE I	464	29
KIBWORTH COMPUTER	483	53	SYSTEM SCIENCE II	493	61
KJD	487	56	SYSTEM STAR	449	9
LAHEY	486	56	UNIPALM	475	47
LBMS	450	11	USA SOFTWARE	461	27
LPA	463	29	USER FRIENDLY	482	53
MAGNIFIEYE	448	6	VEERMUIS	451	13
MICROCOSM	477	48	ZORTECH I	499	75
MICROFT TECHNOLOGY	507	86	ZORTECH II	500	77

As featured in
The Guardian

STOB - Auntie Verity's Hardware Help

'Wouldn't it be a great idea', mused Watts the Editor, 'if somebody wrote us a little hardware column.' In the ensuing general rush for the door, Verity was the one who tripped up on the editorial carpet.

Project 1 Adding extra RAM to your PC

1. Inspect the machine's rear panel, ascertaining the number and type of screws securing casing. You should find eight (8) flat-bladed type screws, many with those crinkly washer things.

2. Inspect your only screwdriver. It is a large Philips-type with all the pointy bits burred off.

3. Borrow your assistant's Swiss penknife and, using the end of the bottle-opener, remove screws 1 to 3.

4. Screw 4 was put on with a hydraulic screw tightener. Spend several minutes ruining its top. At the critical moment, as you give it one last go, the penknife will suddenly fold itself up, nipping the flesh of your right hand. Drop penknife.

5. Borrow the toolset from your Manager's BMW. Use the screwdriver found therein to extract remaining screws. Remove cover, placing screws and washers in a safe place, *viz* the spider plant's water collection saucer.

6. Unscrew a metal blanking strip at the end of a free slot. The point at which the strip may be eased clear is reached when the screw you are turning drops out of its hole and falls somewhere inside the machine.

7. Attempt to retrieve screw using Philips screwdriver and a blob of Blu-Tak.

8. Attempt to retrieve screw and blob of putty by holding machine upside-down and gently shaking it. Screw and VGA card fall out. Screw rolls out of sight.

9. Replace VGA card.

10. Insert RAM card in slot, and secure using a screw from spider plant's dish. This may be quite hard, as the screw is too big for the hole, so be prepared to be firm.

11. Turn on machine. Power On Self Test (POST) reports error in CMOS equipment configuration. Spend 20 minutes hunting for the floppy disk containing SETUP program.

12. Replace case and remaining six screws (another casualty lost down back of radiator) plus any crinkly washers that you can be bothered with. Switch on reassembled machine. Nothing: no power supply fan, no screen, nothing. Chuck it in for the day.

Project 2 Connecting two RS-232 devices

1. Examine devices, and determine that you need a 9-pin male D-type plug to a 25-pin female D-type plug serial cable.

2. Examine your stock of spare cables. You have three (3) mains power cables (none with plugs), one (1) PC to Centronics parallel cable (the printer end is bust, and the wires inside are spilling out like multicoloured intestines), one (1) telephone cable (from that modem

that John bought that never worked properly) and (1) one car jump lead (source unknown).

3. Give assistant a tenner from petty cash and send him out to Shop to acquire requisite connectors. While he is gone, cut connector off the end of a power cable. Peel back 20 mm of outer insulation. Separate inner strands, and baring 3mm of copper.

4. One hour later, assistant returns with one (1) 9-pin female plug and one (1) 25-pin male plug. Ask him what he thinks he is playing at. He explains that he always has a difficulty keeping track of which is which.

5. Deliver to assistant short, sarcastic lecture about Birds, Bees and D-type connectors, which includes a useful *aide memoire*.

6. Suddenly remember that Bill, who has the same setup next door, is out this afternoon. Sneak into his office and steal his cable.

Project 3

Repairing a keyboard which has had a cup of coffee spilled on it

1. Exercising extreme care, remove the four retaining screws from the bottom of the keyboard.

2. It's important to keep your workspace tidy. Sweep up the assortment of springs, clips and rubber bits that have fallen out and throw them away.

3. Phone supplier and order new keyboard.

Opportunities for Software Professionals

SYSTEMS PROGRAMMER

BERKSHIRE £20k

1-2 years COBOL experience within software tool development or applications programming. Knowledge of PC DOS & OS/2 with presentation manager. To provide product development & support skills for important COBOL tools. Ref: 01/91/AJW

SYSTEMS PROGRAMMER

BERKSHIRE £13-15k

To provide programming & maintenance support for end-user products in the IBM PC market. Experience of 8086/ 80286/ 80386, Assembler, DOS or OS/2 for creation of COBOL code generator

Ref: 01/91/BJW

SENIOR SOFTWARE ENGINEER

HAMPSHIRE c£18k

Experience of real-time programming in C & Assembler with knowledge of MS-DOS & UNIX. Conversant with use of logic analysis & in-circuit emulators. To join a team engaged in design & development of software modules for use in fixed control applications
Ref: 01/91/CJW

SENIOR ANALYST PROGRAMMERS/ANALYST PROGRAMMERS

BERKSHIRE £14-20K

Part of project teams covering system development (Software, hardware & telecommunications & training, communications, procedures, documentation). IBM 3090/300J mainframe environment running CICS under MVS/ESA operating system with DB2 database via X25 network. CSP used for development work, FOCUS for report writing. Minimum 2 months experience as analyst/programmer on IBM platform. Degree qualified.

Ref: 01/91/GJW & HJW

DOCUMENTATION TECHNOLOGY PROGRAMMER

BERKSHIRE £Neg

Responsible for designing & programming components for workbench product to enable users to create documentation & tutorials from applications. Products include graphical user interfaces & object orientation. Would suit COBOL or experienced programmer with interest in text processing/DTP.
Ref: 01/91/DJW

DATABASE CONSULTANT

SURREY £Neg

To provide expertise to projects in specialist software engineering & database matters, particularly state of the art command databases. Specific help in the use of Yourdon SASD & object oriented design. Knowledge of structured techniques useful.
Ref: 01/91/EJW

SENIOR SYSTEMS/SOFTWARE ENGINEERS

SURREY £Neg

To manage or participate in study, project, or BID activities & formulate system & software proposals relating to combat, command & sensor or weapons systems. Knowledge of structured system design, high level languages (esp Ada) & Yourdon RTSASD desirable.
Ref: 01/91/FJW

PRODUCT LEADER

BERKSHIRE £21-28k + Car

To be responsible for all aspects of system development of the job management system, including project planning & control. Liaison with variety of users & external suppliers. Present development areas include MIS & interfaces between JMS & Ether corporate systems. Sound project management skills required. Familiarity with mainframe systems essential.
Ref: 01/91/IJW

SENIOR SYSTEMS ANALYST

BERKSHIRE £22k - 24k

A degree plus a minimum of 8 years experience desirable in civil engineering systems. CSP/FOCUS experience, Artemis/AS project management system. IBM mainframe environment, PC's & or prime systems. IBM 3090 environment with MVA/ESA operating systems. Also use of CICS, TSC, & COBOL2 & GPG.
Ref: 01/91/JJW

SUPPORT CONSULTANT

BERKSHIRE £Neg

Responsible for providing a range of support services to users, resellers & internal sales. Expected to be a source of expert knowledge on the use & application of time specific product range. Experience of post-sales support, minimum 4 years PC environment plus MS-DOS, Paradox, Lotus 123 (or Quattro) PC networks, especially Novell & 3COM, UNIX or VAX systems.
Ref: 01/91/KJW

SYSTEMS PROGRAMMER

BERKSHIRE TO £17K

Create new Run Time Systems using 8086/80286/ 80386 assembler and provide maintenance support to all OEM customers. Additional experience of three of the following is essential:- DOS, OS/2, COBOL, C, FLEX, UNIX or VM.
Ref: 01/91/MJW

KNOWLEDGE ENGINEERS

HOME COUNTIES

Rapidly expanding Expert Systems Software house requires young developers to design and implement projects. 1 year's experience minimum of any AI language or C, and UNIX etc.
Ref: 01/91/LJW

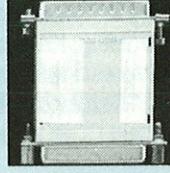
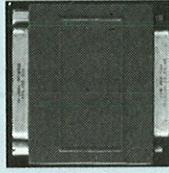
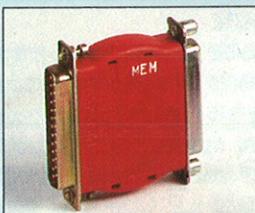
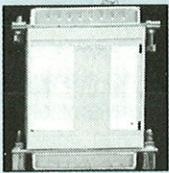
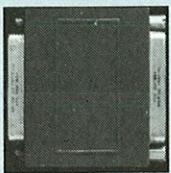
For further information on these or related positions call JEREMY WILLAN or SARAH HOLTHAM on (0734) 774234 or (0604) 33195 after 7p.m. FAX: (0734) 772773
Or write in confidence to CPS at:-

**63 Peach Street
Wokingham
Berkshire RG11 1XP**



SOFTWARE PROTECTION!

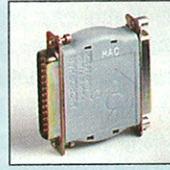
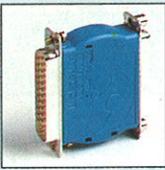
Can You Spot The Difference?



- High-level security keys
- Assembler-based, customer specific, encrypted interrogation routines
- Over 70 languages supported
- Compatibility - due to 8 years experience & 600,000+ keys sold
- Reliable on-going support

- Physically unique keys for each customer
- MS-DOS, OS/2, UNIX, XENIX, WINDOWS 2 & 3, "MACINTOSH"
- Transparent operation with most peripherals on market
- Free language updates
- Parallel, Serial, Mac ports

OVER 3,100 CUSTOMERS WORLDWIDE HAVE!



Electronic Key

The ideal device for identically produced software packages. Uniquely wired with customer code and a software code. Uses Assembler based program, decryption interface and random values.

1 Word Memory Key

Custom hardware wiring allows the developer total control over information stored in the key. 2 bytes of memory allows several packages to be protected with just the one key.

31 Word Memory Key

For multiple software protection schemes. 31 words of 16 bits of non-volatile dynamically programmable memory. Its capacity to store information provides virtually limitless power. Flexible protection scheme can be modified on-site during operation of software package.

Micro Processor Key

Provides the ultimate in software security. Not tied to any language or O/S. 8 bit microprocessor powers from RS-232 level. Requires no power supply. For PC terminals, minis, & others using RS 232 C comms. Used on workstations. This key is effectively a computer.

Macintosh Key

Extremely powerful & customised protection for the Mac. 31 words available for random storage. Providing unequalled protection the Macintosh Memory Key connects to the SCSI 25 pin port and operates transparently.

MICROPHAR... The European Standard!

For a no-nonsense informative discussion on how our dongles can better protect your profits, please call us on:
091-378 9191

CLEARSOFT

CLEARSOFT, Littleburn Ind. Estate,
Langley Moor, Durham DH7 8HG.
Telephone: (091) 378 9191 Fax: (091) 3789393

Please send me full details on your protection systems.

Name:

Position:

Address:

Telephone:

Fax:

Signature:

Clearsoft, Littleburn Ind Est,
Langley Moor, Durham, DH7 8HG
Tel: 091 3789191 Fax: 091 3789393